

**Universidad de Cienfuegos “Carlos Rafael Rodríguez”
Facultad de Informática**



**Tesis presentada en opción al grado de máster en Nuevas Tecnologías para
la Educación**

TÍTULO

**Herramienta Informática de apoyo al Proceso Docente Educativo de la
disciplina *Matemática General* en la carrera *Ingeniería Informática:*
*Matemática III***

AUTOR

Juan Felipe Medina Mendieta

TUTORES

**Narciso Rubén de León Rodríguez
Roberto Fuentes Garí**

2010

Declaración de autoría

Declaro que soy el único autor de este trabajo y autorizo a la Facultad de Informática en la Universidad de Cienfuegos “Carlos Rafael Rodríguez”, para que hagan uso de esta tesis de maestría.

Para que así conste firmamos la presente a los ____ días del mes de ____ del ____.

Nombre completo del autor

Nombre completo del primer tutor

Nombre completo del segundo tutor

Los abajo firmantes certificamos que el presente trabajo ha sido revisado según acuerdo de la dirección de nuestro centro y el mismo cumple los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma Tutor

Firma ICT

Firma Vicedecano

Resumen.

La presente investigación, titulada *Herramienta Informática de apoyo al Proceso Docente Educativo de la disciplina Matemática General en la carrera Ingeniería Informática: Matemática III*, es el resultado de llevar a la par dos aspectos importantes en la formación del ingeniero de la especialidad antes mencionada: utilización de asistentes matemáticos y utilización de software libre.

La utilización de asistentes matemáticos responde directamente al cumplimiento de los objetivos del proceso docente educativo (PDE) de la asignatura Matemática III, planteados en el Plan de Estudios D, en la carrera Ingeniería Informática y la utilización de software libre es una directriz estratégica en nuestro país en la actualidad.

En la investigación se desarrolla una herramienta informática utilizando el asistente matemático y software libre Maxima, que cuenta fundamentalmente de: páginas web, en la que se explican todas las funcionalidades que ya tiene el sistema en relación con los contenidos, organizados por temas de la asignatura Matemática III; y un fichero de instrucciones con otras funcionalidades (las cuales son de igual forma explicadas en las páginas web) que son programadas utilizando el lenguaje de Maxima, de manera que se completan todos los contenidos de la asignatura.

Índice de contenido.

Introducción.....	1
Capítulo I: Marco teórico.	7
1.1 Introducción.....	7
1.2 Descripción de los antecedentes.	7
1.2.1 Importancia de la carrera Ingeniería Informática en la actualidad.	7
1.2.2 Tendencias actuales de la Educación Superior Cubana.	9
1.2.3 ¿Por qué estudiar Matemáticas en la carrera Ingeniería Informática? ..	10
1.2.4 Necesidad de que los estudiantes de informática se relacionen con sistemas informáticos que contribuyan al desarrollo de sus habilidades.	11
1.2.5 Asistentes matemáticos.....	12
1.2.6 Importancia de utilizar software libre en la docencia.	13
1.2.7 Proceso de emigración a software libre en Cuba.	15
1.3 Matemática III.....	16
1.4 Descripción de las herramientas libres disponibles para desarrollar la propuesta.....	17
1.4.1 Macaulay2.	18
1.4.2 Singular	18
1.4.3 Maxima	18
1.4.4 Axiom.....	19
1.4.5 Yacas.....	20
1.4.6 GNU Octave	20
1.4.7 Euler	21
1.4.8 R.....	21
1.5 Herramienta libre seleccionada para desarrollar la propuesta.	22
1.6 Metodología de desarrollo.....	22
1.6.1 Metodologías Estructuradas y Metodologías Orientadas a Objetos. .	23
1.6.2 Metodologías Tradicionales y Metodologías Ágiles.	23
1.6.3 Metodología seleccionada para desarrollar la propuesta.	24
1.7 Conclusiones.....	24
Capítulo II. Descripción y construcción de la solución propuesta.....	25
2.1 Introducción.....	25
2.2 Concepción general del sistema.	25
2.3 Requerimientos mínimos del sistema.....	27

2.4	Análisis y diseño estructurado.....	28
2.4.1	Principios básicos del análisis y diseño estructurado.	28
2.4.2	Diagramas de flujo de datos (DFD).	29
2.4.3	Representación lógica en los procesos computacionales. Lenguaje natural estructurado (LNE).	32
2.4.4	Diccionario de datos (DD).	38
2.4.5	Relaciones entre las herramientas del análisis y diseño estructurado... ..	58
2.5	Conclusiones.....	59
Capítulo III. Consideraciones para la utilización de la herramienta propuesta en la asignatura Matemática III		60
3.1	Introducción.....	60
3.2	Plan de Estudios D de la asignatura Matemática III en la carrera Ingeniería Informática.	60
3.3	Derive vs Maxima en la asignatura Matemática III.....	62
3.4	Ventajas de la utilización de la herramienta propuesta en la asignatura Matemática III.	65
3.5	Herramienta propuesta como instrumento que favorece el PDE de la asignatura Matemática III.....	66
3.6	Conclusiones.....	70
Conclusiones.....		71
Recomendaciones.....		72
Bibliografía.		73
Anexos.		76

Índice de tablas.

Tabla 1. Definición de flujo de datos. Datos tipo 1.	39
Tabla 2. Definición de flujo de datos. Datos tipo 2.	40
Tabla 3. Definición de flujo de datos. Datos tipo 3.	40
Tabla 4. Definición de flujo de datos. Datos tipo 4.	41
Tabla 5. Definición de flujo de datos. Solución.	42
Tabla 6. Definición de flujo de datos. Respuesta.	43
Tabla 7. Definición del elemento de datos. Función.	43
Tabla 8. Definición del elemento de datos. Variable.	43
Tabla 9. Definición del elemento de datos. Término enésimo.	44
Tabla 10. Definición del elemento de datos. Variable de la serie.	44
Tabla 11. Definición del elemento de datos. Valor numérico donde comienza la serie.	45
Tabla 12. Definición de Tarea Funcional. Resolver EDO variables separables. ...	46
Tabla 13. Definición de Tarea Funcional. Resolver EDO homogéneas.	46
Tabla 14. Definición de Tarea Funcional. Resolver EDO exacta.	47
Tabla 15. Definición de Tarea Funcional. Resolver EDO reducible a exacta.	48
Tabla 16. Definición de Tarea Funcional. Resolver EDO lineal.	49
Tabla 17. Definición de Tarea Funcional. Resolver EDO Bernoulli.	50
Tabla 18. Definición de Tarea Funcional. Criterio serie término enésimo.	51
Tabla 19. Definición de Tarea Funcional. Criterio serie comparación.	52
Tabla 20. Definición de Tarea Funcional. Criterio serie raíz.	53
Tabla 21. Definición de Tarea Funcional. Criterio serie cociente.	54
Tabla 22. Definición de Tarea Funcional. Criterio serie Raabe.	55
Tabla 23. Definición de Tarea Funcional. Criterio serie Gauss.	56
Tabla 24. Definición de Tarea Funcional. Criterio serie convergencia.	57
Tabla 25. Derive vs. Maxima	65

Introducción

El avance de las Tecnologías de la Información y de las Comunicaciones (TIC) ha influido significativamente en la sociedad a escala mundial justificando la tesis que plantea que *nos encontramos en lo que se ha dado a llamar sociedad de la información y del conocimiento*. Nuestro país no ha estado al margen de este hecho llevando a cabo cambios y medidas orientadas hacia el desarrollo de las TIC.

El desarrollo gradual de las TIC en la sociedad cubana ha provocado profundas transformaciones. Una de ellas se ha llevado a cabo en los medios de enseñanza en los diferentes niveles. Esto ha implicado cambios importantes en los roles tradicionales del profesor y el estudiante. Dadas estas nuevas condiciones la educación superior ha tenido la necesidad de una nueva concepción curricular.

Esta nueva concepción de la educación superior cubana plantea la disminución de la actividad presencial y el incremento del auto aprendizaje de los estudiantes, a partir de la introducción de nuevos medios en el proceso de formación, como las Tecnologías de la Información y las Comunicaciones. (Cañedo, 2008)

La carrera Ingeniería Informática materializa en el nuevo Plan de Estudios D de julio de 2007 estas tendencias: “Incrementar el estudio independiente, asistido por las Tecnologías de la Información y las Comunicaciones (TIC)” (MES, 2007 pág. 16).

El estudiante de esta carrera, adaptado a las condiciones actuales debe, como parte de su formación, vincularse con diferentes software que contribuyan tanto al desarrollo de sus habilidades, como a la formación de una cultura informática que le permita valorar el funcionamiento de los sistemas y adquirir experiencia (MES, 2007). Cobra gran importancia, en particular, la utilización de software que permitan reutilizar el código y modificarlo.

No sólo es de interés del estudiante de informática el vínculo con diferentes sistemas: “todas las disciplinas de la carrera se complementan en la formación de los estudiantes en las diferentes dimensiones.” (MES, 2007 pág. 50). “Las diferentes disciplinas aportan los elementos básicos para el trabajo en el campo de la computación,...” (MES, 2007 pág. 50).

Es por ello que disciplinas como Matemática General buscan alternativas para lograr un mayor vínculo con la especialidad y así contribuir a la formación integral de los estudiantes. La utilización de asistentes matemáticos es un buen ejemplo que lo expresa.

“El propósito de la enseñanza de la disciplina Matemática consiste en adiestrar a los estudiantes en la utilización de los distintos métodos analíticos y aproximados, en el uso de asistentes matemáticos y en la implementación de esquemas de cálculo en máquinas computadoras, desarrollando así su pensamiento lógico, heurístico y algorítmico.” (MES, 2007 pág. 105)

“La disciplina tendrá cinco asignaturas semestre, tres en primer año y dos en segundo” (MES, 2007 pág. 114). En primer año Álgebra, Matemática I y Matemática II y en segundo año Matemática III y Matemática IV. “La computación estará presente en cada una de las asignaturas que componen la disciplina,...” (MES, 2007 pág. 114).

Existen disímiles asistentes matemáticos que pueden ser utilizados (y son utilizados) en las asignaturas de la disciplina Matemática General. Estos pueden ser clasificados en software libre o privativo atendiendo al tipo de licencia que tengan y en simbólicos o numéricos de acuerdo al tipo de procesamiento que lleven a cabo (Asistentes matemáticos y Software libre, 2008). Investigaciones realizadas por Galván en el año 2005 y Pardini en el 2007 apoyan la idea de utilizar software libre en la docencia, debido a sus ventajas técnicas, morales y pedagógicas por sobre sus contrapartes privativas.

Software libre es aquel que permita: ejecutar el programa con cualquier propósito, acceder al código fuente para estudiar su funcionamiento y adaptarlo a las necesidades de cada uno, redistribuir copias, mejorar el programa y luego publicarlo para el bien de toda la comunidad. (Stallman, 2004)

El modelo de software privativo trae consigo varias desventajas: restricciones impuestas para su uso, formación basada en la excesiva dependencia de una única herramienta. En un ámbito educativo una herramienta privativa pudiera conllevar a problemas de tipo éticos: no basar la educación en pilares como la libertad, el conocimiento, la solidaridad y la colaboración y el hecho de no educar a

la sociedad con información ajena a la manipulación, con espíritu de cooperación y con libertad. (Asistentes matemáticos y Software libre, 2008).

He aquí la importancia que se ha dado al software libre debido que permite tener control total sobre el software: estudiarlo, modificarlo, mejorarlo y, fundamentalmente, compartirlo. Esto último posibilita integrar y potenciar valores éticos que son importantes en la formación de los estudiantes universitarios.

En la carrera Ingeniería Informática en particular, es muy importante que los estudiantes utilicen software libre que les permitan acceder al código fuente, para estudiarlo, saber cómo funciona y poder hacer modificaciones, crear sus propios algoritmos y hasta tener la posibilidad de publicar sus mejoras. Todo esto unido a que el Ministerio de Educación Superior (MES) está involucrado en el proceso de emigración a software libre, lo cual responde a una política del país, hacen necesario buscar alternativas libres para los sistemas privativos que los estudiantes utilizan en las diferentes asignaturas.

Actualmente no se utiliza en la Universidad de Cienfuegos ningún asistente matemático basado en la filosofía de software libre que ayude a cumplir los objetivos del PDE planteados en el Plan de Estudios D para la disciplina Matemática General y, particularmente, para la asignatura Matemática III. Es por esto que se plantea el siguiente problema:

Problema científico:

Necesidad de contar con una herramienta informática, basada en la filosofía de software libre, para apoyar el PDE de la asignatura Matemática III de la disciplina Matemática General en la carrera Ingeniería Informática.

Objeto de investigación:

El PDE de la asignatura Matemática III de la disciplina Matemática General en la carrera Ingeniería Informática, así como los software libres existentes que permitan desarrollar una herramienta informática para esta disciplina.

Campo de acción:

Los componentes del PDE de la asignatura Matemática III de la Disciplina Matemática General en la carrera Ingeniería Informática.

Objetivo general de la investigación:

Desarrollar una herramienta informática basada en la filosofía de software libre para apoyar el PDE de la asignatura Matemática III de la Disciplina Matemática General en la carrera Ingeniería Informática.

Objetivos específicos:

1. Analizar el modelo del profesional de Ingeniería Informática (Plan de Estudios D)
2. Analizar el programa de la disciplina Matemática General en la carrera Ingeniería Informática.
3. Analizar el programa de la asignatura Matemática III.
4. Analizar software libres existentes que permitan desarrollar una herramienta informática para la disciplina Matemática General en la carrera Ingeniería Informática.
5. Diseñar e implementar una herramienta informática para la asignatura Matemática III.
6. Validar la herramienta informática.

Idea a defender:

Si se utiliza una herramienta informática basada en la filosofía de software libre como apoyo a la asignatura Matemática III de la disciplina Matemática General en la carrera Ingeniería Informática, entonces se favorece el cumplimiento de los objetivos del PDE de esta asignatura.

Tareas científicas:

1. Síntesis de aspectos relacionados con:
 - El programa de la asignatura Matemática III de la disciplina Matemática General en la carrera Ingeniería Informática.
 - Software libres existentes que permitan desarrollar una herramienta informática para la disciplina Matemática General en la carrera Ingeniería Informática.
2. Análisis y diseño de la herramienta.
3. Desarrollo de la herramienta informática.
4. Preparación de la herramienta.
5. Validación de la herramienta.
6. Documentación de la investigación.

Métodos:

En cuanto a métodos teóricos, se utilizó el análisis histórico-lógico para resumir y precisar la información recopilada sobre los software libres existentes.

El método de análisis y síntesis se empleó para conocer los objetivos del PDE de la asignatura Matemática III en la carrera Ingeniería Informática planteados en el Plan de Estudios D y la influencia de la utilización de asistentes matemáticos en la misma.

El método de inducción-deducción se usó para garantizar un resultado de calidad en la herramienta informática a proponer.

En cuanto a los métodos empíricos se utilizaron la entrevista individual y grupal para recopilar información de los expertos.

Aporte práctico:

El desarrollo de una herramienta informática propia basada en la filosofía de software libre como apoyo a la asignatura Matemática III de la disciplina Matemática General de la carrera Ingeniería Informática enriquece el PDE de la misma, incide positivamente en la formación del profesional y responde a la necesidad de utilizar software libre, lo cual forma parte de una política del país y del Ministerio de Educación Superior.

Caracterización de los capítulos del informe:

El presente documento está estructurado en tres capítulos, conclusiones y recomendaciones. El contenido de cada capítulo es:

Capítulo I: Marco teórico.

Se presenta una descripción detallada de los antecedentes de la investigación, se realiza una descripción de las herramientas libres disponibles para desarrollar la propuesta y por último se explica la herramienta libre escogida justificando su selección.

Capítulo II: Descripción y construcción de la solución propuesta.

Se describe la concepción general del sistema propuesto y los requerimientos mínimos necesarios para el adecuado funcionamiento del mismo. Se realiza el análisis y diseño estructurado del sistema que integra: diagrama de flujo de datos, representación lógica de los procesos computacionales (lenguaje natural estructurado), diccionario de datos y la interrelación entre estas herramientas.

Capítulo III: Consideraciones para la utilización de la herramienta propuesta en la asignatura Matemática III.

Se aborda parte del desarrollo del PDE de la asignatura Matemática III de la disciplina Matemática General en la carrera Ingeniería Informática. Se hace un análisis del programa Derive utilizado históricamente y actualmente en la asignatura como herramienta computacional, donde se describen sus aspectos significativos. Se justifica la utilización de Maxima como alternativa libre de Derive a través de la herramienta propuesta en esta investigación.

Capítulo I: Marco teórico.

1.1 Introducción.

Este capítulo, como parte de la fundamentación teórica del tema de la investigación, presenta una descripción detallada de los antecedentes de la investigación en la que se trata: la importancia de la carrera Ingeniería Informática en nuestros días, las tendencias actuales de la educación superior cubana, la importancia del estudio de las matemáticas en la carrera Ingeniería Informática, la necesidad de que los estudiantes de informática utilicen diferentes software en su formación, los asistentes matemáticos, la importancia de utilizar software libre en la docencia y el proceso de migración a software libre en Cuba. Se realiza, además, una descripción de las herramientas libres disponibles para desarrollar la propuesta y, por último, se explica la herramienta libre escogida justificando su selección.

1.2 Descripción de los antecedentes.

1.2.1 Importancia de la carrera Ingeniería Informática en la actualidad.

A partir de mediados del siglo XX hasta la actualidad ha evolucionado la ciencia y las TIC, no solo a nivel mundial, sino también en la sociedad y la economía cubanas, en las que la informática ha continuado introduciéndose y desarrollándose.

La especialidad de Ingeniero en Sistemas Automatizados de Dirección Técnico Económico (SAD-TE) creada en 1976 surge debido a la cantidad de máquinas computadoras electrónicas, y otros medios técnicos de computación, que se preveía fuesen introducidos paulatinamente en ministerios, empresas y unidades presupuestadas con el fin de hacer más eficiente la dirección y la gestión productiva y de servicio de estas entidades, incidiendo de forma positiva en la economía nacional. Esta especialidad concebida con un perfil amplio pretendía abarcar todo lo que tenía que ver con la automatización de los sistemas de

información y de toma de decisiones para la gestión y los procesos tecnológicos. (MES, 2007 pág. 13)

En las décadas de los 80 y 90, debido a diferentes acontecimientos de carácter global y nacional (Ejemplo: Popularización y comercialización a gran escala de las computadoras personales), nuestro país acelera el desarrollo de la informática y las comunicaciones.

En la Resolución Económica del V Congreso del Partido Comunista de Cuba, de octubre de 1997, queda declarada explícitamente la importancia actual de la informática y la industria del software al expresarse:

- “En particular las industrias ligera, de materiales de construcción, otras ramas de la industria básica, las actividades industriales de diferentes organismos, y sideromecánica y electrónica, incluida la del software, deberán trabajar con calidad, por incrementar la sustitución de importaciones, y desarrollar la exportación de líneas de elevada eficiencia.

- El país debe encaminarse resueltamente a la modernización informática mediante un programa integral que involucre a las organizaciones que deben proveer los recursos materiales, financieros e intelectuales y a las entidades económicas, políticas y sociales que deben traducirlos en más y mejores productos y servicios.” (MES, 2007 pág. 14)

Lo anterior fundamenta el estudio de la carrera Ingeniería Informática en nuestro país. A partir del 2002 se produce un importante cambio en el escenario de la formación de ingenieros informáticos. En este nuevo escenario hay tres formas diferentes de cursar el plan de estudio:

- El Curso Regular Diurno (CRD).
- El plan de Continuidad de Estudios para maestros de Computación Básica de primaria que se cursa en las Sedes Universitarias Municipales.
- El plan de la Universidad de las Ciencias Informáticas, con régimen de internado para estudiantes procedentes de todo el país, que posee condiciones productivas que posibilitan una muy alta dedicación al estudio y al trabajo. (MES, 2007 pág. 16).

Actualmente el estudio de esta especialidad tiene sus particularidades, regidas fundamentalmente por las tendencias de la educación superior cubana.

1.2.2 Tendencias actuales de la Educación Superior Cubana.

El vertiginoso avance de las TIC ha influido significativamente en la sociedad a escala mundial. A consecuencia de esto se plantea que nos encontramos en lo que se ha dado a llamar *sociedad de la información y del conocimiento*. Nuestro país no ha estado al margen de este hecho, llevando a cabo cambios y medidas orientadas al desarrollo de las TIC.

Este desarrollo, en la sociedad cubana, ha provocado profundas transformaciones en los métodos de enseñanza en los diferentes niveles. Esto ha implicado cambios importantes en los roles tradicionales del profesor y el estudiante. Dadas estas nuevas condiciones, la educación superior ha tenido la necesidad de una nueva concepción curricular.

Entre las tendencias de la educación superior cubana está la disminución de la actividad presencial de los estudiantes, a partir de la introducción de nuevos métodos en el proceso de formación que centran su atención principal en el auto aprendizaje de los estudiantes. En este sentido las TIC desempeña un importante papel. (Cañedo, 2008)

“Como consecuencia de un amplio y generalizado empleo de las TIC se producen importantes transformaciones en el proceso de formación, que han de expresarse fundamentalmente en el cambio en los roles de profesores y estudiantes y en el empleo de métodos más colaborativos, que permitan mayor asincronismo en el proceso de formación. En relación con ello se debe atender prioritariamente aspectos tales como la utilización de software profesionales, el empleo generalizado de plataformas interactivas de trabajo que fortalezcan el autoaprendizaje de los estudiantes y la introducción de prácticas de laboratorios virtuales” (Cañedo, 2008).

La carrera Ingeniería Informática materializa en el nuevo Plan de Estudios D estas tendencias de la educación superior cubana: “Incrementar el estudio independiente, asistido por las Tecnologías de la Información y las Comunicaciones (TIC).” (MES, 2007 pág. 16)

En el caso de la disciplina Matemática Básica de la carrera Ingeniería Informática también se encuentran presentes estas tendencias y así está planteado en sus Indicaciones Metodológicas y de Organización del Plan D.

Refiriéndose al desarrollo de las asignaturas se plantea: “Es necesario diseñar adecuadamente el modo de trabajo de la asignatura, el sistema de evaluación y el sistema de trabajo independiente. Se debe tender a:

- Incrementar el uso de los medios de cómputo, la utilización de bibliotecas y la creación de programas sencillos.
- Plantear problemas al estudiante en los que, de modo independiente, debe realizarse la creación, adaptación e interpretación de un modelo, su solución por computadora, el análisis de los errores y la validación del resultado.” (MES, 2007 pág. 115)

1.2.3 ¿Por qué estudiar Matemáticas en la carrera Ingeniería Informática?

Luego de haber analizado la importancia del estudio de la carrera Ingeniería Informática en nuestro país regido fundamentalmente por las tendencias actuales de la educación superior cubana, es necesario, producto al peso que tiene en la investigación, fundamentar la importancia de estudiar matemática en esta carrera.

Las matemáticas constituyen asignaturas básicas que preparan y forman al estudiante. Objetivos instructivos del primer año de la carrera plantean: “desarrollar el pensamiento abstracto, el análisis lógico y el razonamiento inductivo y deductivo” y “adquirir habilidades elementales en la interpretación de gráficos y otras formas de representación tridimensional” (MES, 2007 pág. 19), lo cual reafirma el planteamiento anterior.

En la carrera Ingeniería Informática, las matemáticas se enmarcan en dos disciplinas: Matemática General y Matemática Aplicada. Matemática General consta de cinco asignaturas: Álgebra, Matemática I, Matemática II, Matemática III y Matemática IV repartidas entre primero y segundo años de la carrera (MES, 2007 pág. 114). Matemática Aplicada consta de tres asignaturas: Probabilidades y Estadística, Investigación de Operaciones y Simulación, repartidas entre tercero y cuarto año de la carrera. (MES, 2007 pág. 130)

“La disciplina Matemática General, es aquella en que se desarrollan los fundamentos de la formación de un especialista en Ingeniería Informática, dado que todo ingeniero considera representaciones técnicas y científicas en términos matemáticos con los cuales reflejan los rasgos cuantitativos de los fenómenos que estudia. De tal modo, el objetivo de esta disciplina es lograr que el ingeniero informático domine el aparato matemático que lo haga capaz de modelar y analizar los procesos técnicos, económicos, productivos y científicos, utilizando en ello, tanto métodos analíticos como aproximados y haciendo uso eficiente de las técnicas de cómputo.” (MES, 2007 pág. 105)

“El propósito de la enseñanza de la disciplina Matemática consiste en adiestrar a los estudiantes en la utilización de los distintos métodos analíticos y aproximados, en el uso de asistentes matemáticos y en la implementación de esquemas de cálculo en máquinas computadoras, desarrollando así su pensamiento lógico, heurístico y algorítmico.” (MES, 2007 pág. 105)

La disciplina Matemática General contribuye de forma general a la formación del ingeniero informático debido a que:

- “Amplía la madurez matemática y la capacidad de trabajo con la abstracción.
- Desarrolla habilidades para la comunicación de propiedades y características de magnitudes en forma gráfica, numérica, simbólica y verbal.
- Contribuye a conformar una cultura científica general e integral actualizada, que toma en cuenta:
 - El uso de la computación en la resolución de problemas,
 - El procesamiento de literatura técnica,
 - El manejo de lenguaje interdisciplinario.” (MES, 2007 pág. 107)

1.2.4 Necesidad de que los estudiantes de informática se relacionen con sistemas informáticos que contribuyan al desarrollo de sus habilidades.

Para el estudiante de informática es importante el hecho de relacionarse con diferentes software que contribuyan tanto al desarrollo de sus habilidades, como a

la formación de una cultura informática que les permita valorar el funcionamiento de los sistemas y adquirir experiencia.

En particular, la utilización de software que permitan reutilizar el código y modificarlo, debido al estándar actual de desarrollo de software en el mundo, es mucho mejor para su formación profesional.

“En las diferentes asignaturas de la carrera deberán utilizarse software de aplicación específicos del perfil del graduado, ya sean paquetes profesionales o desarrollados por el claustro de profesores u otros especialistas del país (...) Estos software serán usados por los estudiantes como apoyos en su labor docente e investigativa, además del uso de entornos de desarrollo de software que ya son más específicos de la profesión.” (MES, 2007 pág. 51)

Partiendo de la premisa de que “todas las disciplinas colaboran en la formación general del egresado” (MES, 2007 pág. 51), se plantea el hecho de que asignaturas básicas como las de Matemática General logren un mayor vínculo con la especialidad y así se contribuya a la formación integral de los estudiantes. En este caso la utilización de asistentes matemáticos brinda esa posibilidad.

1.2.5 Asistentes matemáticos.

“Los mediadores, intérpretes o intermediarios entre el ordenador y las ciencias de las Matemáticas son aquellos programas desarrollados con este fin, es lo que se conoce comúnmente como Asistentes matemáticos o Herramientas matemáticas” (Asistentes matemáticos y Software libre, 2008 pág. 3). Estos se han desarrollado para apoyar el proceso docente educativo y el procesamiento matemático de modelos, en ámbito de proyectos investigativos.

Un programa debe cumplir determinados requerimientos o requisitos para poder ser usado con éxito como herramienta matemática, sobre todo si se enmarca en el ámbito de la educación superior. Entre los requerimientos fundamentales deben señalarse los de cálculos científicos (requerimiento mínimo), cálculos simbólicos y cálculos numéricos y matricial (Estos dos últimos requerimientos en función de las especificidades de cada herramienta) (Galván, 2005 pág. 5).

“Existen múltiples Asistentes Matemáticos (desarrollados unos con fines académicos y otros para apoyar problemas matemáticos) que cumplen con estos

requisitos y cuentan con muchas otras funcionalidades” (Asistentes matemáticos y Software libre, 2008 pág. 3). Tomando en cuenta la licencia bajo la cual se distribuyen, estos pueden clasificarse en: Asistentes matemáticos propietarios (software privativo) y Asistentes matemáticos libres (software libre).

1.2.6 Importancia de utilizar software libre en la docencia.

Se llama software libre a aquél que garantice las siguientes libertades:

- Libertad 0: La libertad para ejecutar el programa con cualquier propósito.
- Libertad 1: La libertad para estudiar el funcionamiento del programa y adaptarlo a las necesidades de cada uno— El acceso al código fuente es condición indispensable para esto.
 - Libertad 2: La libertad para redistribuir copias y ayudar a otros.
 - Libertad 3: La libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad— El acceso al código fuente es condición indispensable para esto (Stallman, 2004 pág. 45).

“El software libre no tiene restricciones en cuanto a distribución de copias se refiere, puede ser modificado sin tener que pedir permiso ni pagar por ello, es adquirido de forma gratis o pagando por su distribución y puede ser copiado, o comercializado.” (Asistentes matemáticos y Software libre, 2008 pág. 7)

Programas que no cumplan con algunas de las cuatro libertades anteriores son denominados software privativo.

Investigaciones realizadas apoyan la idea de utilizar software libre en la docencia, debido a sus ventajas técnicas, morales y pedagógicas por sobre sus contrapartes privativas.

En el año 2005 Rafael Rodríguez Galván, profesor del Departamento de Matemáticas de la Universidad de Cádiz, publica: “Matemáticas y Software libre para la docencia en la Universidad de Cádiz “. En este artículo se concluye que:

- Una formación basada en la excesiva dependencia de una única herramienta comercial, puede llegar, con el tiempo, a ser obsoleta.
- La dependencia de una herramienta privativa en el ámbito educativo conlleva problemas éticos añadidos, puesto que de forma irremisible provoca en el

alumnado la seducción por una marca cuyo precio hará que, en la mayoría de los casos, no pueda ser adquirida legalmente, lo que incita su copia ilegal.

- Al usar en el aula una herramienta con licencia libre, el profesor cuenta con ventajas adicionales a la hora de la planificación y el desarrollo de la asignatura derivadas de tener la garantía de que un programa con software libre podrá ser instalado y usado por sus alumnos en cualquier lugar fuera de la universidad.

- El software libre posibilita que los alumnos puedan acceder al código fuente, a la forma en que está programada la herramienta que están utilizando en clase. Esto aporta el gran valor de poder estudiar la manera en la que un programa de primer nivel implementa en la práctica los algoritmos que han sido estudiados en las clases teórico-prácticas. El poder observar y modificar el código de un programa profesional, utilizado por miles de personas de todo el mundo, constituye una experiencia extraordinariamente gratificante, de gran valor docente, como refuerzo y motivación.

- Utilizar software libre en el aula impulsa valores éticos en los cuales se basa la educación como: la libertad, el conocimiento, la solidaridad y la colaboración (Galván, 2005 pág. 4)

En octubre de 2007 A. Pardini de la Universidad nacional de La Plata publicó: “Fundamentación del uso de software libre en la universidad pública. Enseñando Matemática con herramientas alternativas”. En este artículo se defiende la siguiente idea:

“El avance en la comunidad científica siempre fue acompañado por la libre circulación de las ideas y el conocimiento; entonces, ¿por qué educamos usando herramientas informáticas cuyo modelo de distribución no solo fomenta todo lo contrario sino que tampoco nos permite estudiar su funcionamiento? El ámbito académico supuestamente tiene que favorecer la creatividad, innovación, aprendizaje, cooperación y libre flujo de ideas, sin embargo si enseñamos utilizando herramientas privativas estamos atentando contra estos ideales” (Pardini, 2007 pág. 1).

El modelo de software privativo trae consigo varias desventajas. Una es su licenciamiento, que impide distribuirlo; la universidad quizás pueda costear el pago

de licencias pero un alumno no. Esto conlleva a la piratería, dado que se exige el uso de un programa específico, el alumno debe tenerlo para realizar el estudio independiente y la única forma de lograr esto es obtener copias no legítimas. Otra desventaja, no menos importante, es que no es posible estudiar cómo funciona y por tanto no es posible modificarlo. Una alternativa ante esta situación es el software libre. No sólo simboliza un ahorro económico, también representa el retorno del control hacia los usuarios. Es posible crear herramientas propias y se puede hacer lo que se desee: estudiarla, modificarla, mejorarla, combinarla con otras y, fundamentalmente, compartirla. Esto último permite formar y fomentar valores éticos de gran importancia en los estudiantes universitarios.

1.2.7 Proceso de emigración a software libre en Cuba.

En 2005, Cuba inició su emigración a software libre y plataformas de código abierto en general, después de que el Consejo de Ministros decidiera comenzar paulatinamente el traslado de las instancias de la Administración Central del Estado a la nueva tecnología. Desde entonces quedó orientado que esta “emigración” fuera un proceso continuo y organizado, y se crea el Grupo Ejecutivo Nacional encabezado por el ministro de Informática y las Comunicaciones. (Guzmán, 2009)

En la actualidad varias son las organizaciones involucradas en este proceso de migración entre ellos: los Ministerios de Informática y Comunicaciones (MIC), Educación Superior (MES), Cultura (MINCULT) y la Empresa de Telecomunicaciones de Cuba SA. (ETECSA). La Universidad de las Ciencias Informáticas (UCI) es la rectora en la estrategia cubana para esta migración.

El IV Taller Internacional de Software Libre fue el espacio escogido para presentar la Guía Cubana para el cambio a sistema de código abierto. Este será el documento rector mediante el cual las empresas e instituciones organizarán y desarrollarán su propia migración según las características de cada lugar. Dentro de la estrategia está prevista la introducción paulatina del software libre en todos los niveles de enseñanza y, en particular, los Joven Club de Computación serán el espacio ideal para la capacitación masiva.

A partir de todo lo analizado en este epígrafe se pone de manifiesto la necesidad de buscar alternativas libres para los sistemas privativos que los estudiantes utilizan en las diferentes asignaturas.

1.3 Matemática III.

La asignatura Matemática III se imparte en el primer semestre de segundo año de la carrera Ingeniería Informática. Se puede resumir como la asignatura que trata ecuaciones diferenciales, tipos y métodos analíticos de solución y series.

Según el Plan de Estudios D los objetivos instructivos de esta asignatura son los siguientes:

- “Desarrollar capacidades para caracterizar e interpretar los conceptos y teoremas más importantes relativos a las ecuaciones diferenciales y las Series” (MES, 2007 pág. 126).
- “Resolver modelos que se basen en ecuaciones diferenciales ordinarias y sistemas de ecuaciones diferenciales” (MES, 2007 pág. 126).
- “Modelar fenómenos de la realidad utilizando ecuaciones diferenciales ordinarias y sistemas de ecuaciones diferenciales” (MES, 2007 pág. 126).
- “Aproximar funciones por series, analizar las propiedades de la serie aproximadamente y utilizar la aproximación como instrumento para resolver problemas” (MES, 2007 pág. 126).

Los conocimientos básicos a adquirir:

“Ecuaciones Diferenciales Ordinarias (EDO). EDO de primer orden. EDO lineales con coeficientes constantes. Sistemas de EDO lineales con coeficientes constantes. Métodos de solución...” (MES, 2007 pág. 126)

“Series numéricas. Series de funciones: Series de potencias. Series de Taylor. Series de Fourier y aplicaciones. Elementos de la teoría de análisis de algoritmos” (MES, 2007 pág. 126).

En el contexto que ocupa esta investigación, la asignatura Matemática III está organizada en dos temas:

Tema #1: *Ecuaciones diferenciales ordinarias.*

Tema #2: *Series.*

En la planificación de esta asignatura también se presta gran importancia a la utilización de asistentes matemáticos para contribuir a la formación computacional del estudiante, así aparece reflejado en uno de los objetivos educativos del Plan de Estudios D:

“Contribuir a la formación computacional de los estudiantes mediante la utilización de asistentes matemáticos, la creación de algunos programas y el desarrollo de la capacidad de algoritmizar” (MES, 2007 pág. 126)

El asistente matemático Derive es el que se ha estado utilizando para apoyar el PDE de esta asignatura en la Facultad de Ingeniería Informática de la Universidad de Cienfuegos. Este es un asistente matemático profesional bajo licencia de software privativo especializado en el cálculo simbólico, que permite representaciones gráficas en 2D y 3D con un lenguaje de programación interno para el desarrollo de instrucciones y con amplias posibilidades para la docencia (Wikipedia).

Existen otros asistentes matemáticos profesionales de probada calidad, con amplias posibilidades y que cubren perfectamente las necesidades de esta asignatura como: *Mathematica*, *Maple*, *Matlab*; pero todos estos, al igual que Derive, se encuentran bajo el título de licencia propietario y aunque existen herramientas creadas con su uso, las cuales no tienen necesariamente que atenerse al dogma privado, sí debe cumplirse con una serie de leyes (en dependencia del propietario y de las restricciones que le imponga) que rigen a esta clase de programas. “El utilizar este tipo de software supone el conocimiento previo de la licencia de los mismos, madurez y responsabilidad en su uso” (Asistentes matemáticos y Software libre, 2008 pág. 5)

1.4 Descripción de las herramientas libres disponibles para desarrollar la propuesta.

A continuación se describen algunos asistentes matemáticos con licencia libre que fueron analizados para desarrollar la herramienta propuesta, así como la bibliografía donde se puede encontrar mayor información de ellos.

1.4.1 Macaulay2.

Macaulay2 es un sistema de álgebra computacional orientado a la geometría algebraica y al álgebra conmutativa. Este software está disponible en código fuente para portarlo, y en forma compilada para Linux, Sun OS, Solaris, Windows. (Wikipedia Macaulay)

1.4.2 Singular

Es un sistema de álgebra computacional para cálculos polinómicos con énfasis especial en las necesidades del álgebra conmutativa, de la geometría algebraica, y de la teoría de singularidades (“Singular-Wikipedia, la enciclopedia libre,” s.d.), (H. Schönemann Lossen, 2006).

Características:

- Principales objetos de cómputo: polinomios y módulos sobre una gran variedad de anillos.
- Gran variedad de algoritmos implementados, escrito en C/C++, o como bibliotecas.
- Uso de lenguaje de programación similar a C.
- Documentación extensa.
- Disponible para la mayoría de las plataformas: Unix (HP-UX, SunOS, Solaris, Linux, AIX), Windows, Macintosh (Lossen, 2006), (Wikipedia Singular).

1.4.3 Maxima

Sistema de álgebra computacional con un motor de cálculo simbólico escrito en lenguaje *Lisp* publicado bajo licencia GNU GPL. Cuenta con un amplio conjunto de funciones para hacer manipulación simbólica de polinomios, matrices, funciones racionales, integración, derivación, series de *Taylor*, transformadas de *Laplace*, resolver ecuaciones diferenciales y sistemas, calcular límites, trabajar con vectores, listas y tensores, manejo de gráficos en 2D y 3D, manejo de números reales, expansión en series de potencias y de *Fourier*. Permite exportar los resultados a Tex y su funcionamiento está muy bien documentado.

Maxima es la continuación libre del proyecto Macsyma (*MAC's SYmbolic MANipulator*), iniciado en los 60 en el prestigioso MIT (Massachusetts Institute of

Technology), este constituyó el primer proyecto con éxito en la automatización de operaciones matemáticas de tipo simbólico. En este sentido, puede considerarse el padre de todos los programas de cálculo simbólico actuales (en especial Mathcad y Maple) (Pardini, 2007 pág. 3), (Wikipedia Maxima).

Características:

- Cuenta con el apoyo de una comunidad de desarrollo muy dinámica, que trabaja constantemente para su actualización, mejora y soporte.
- Disponible para distintas plataformas, tales como GNU/Linux y MS Windows.
- Utiliza la herramienta gnuplot para la representación de gráficos 2D y 3D.
- Cuenta con manuales y documentación que pueden descargarse libremente de internet.
- Incorpora un completo lenguaje de programación propio, además del *Lisp*.
- Cuenta con diferentes interfaces con los usuarios: de texto por consola a modo de comandos, *xMaxima* y a modo de ventanas, *wxMaxima*.

xMaxima es una interfaz gráfica a modo de comandos que se incluye por defecto con el programa Maxima. Ha sido la interfaz gráfica oficial de Maxima.

wxMaxima es una interfaz gráfica a modo de ventanas que pretende facilitar el uso de Maxima a través de una serie de menús y barras de botones (Galván, 2005 págs. 21-22).

1.4.4 Axiom

Es un asistente de cálculo simbólico por excelencia, sólido, potente y expansible. Es un sistema abierto, modular y diseñado para soportar un gran número de nuevas características con un mínimo incremento en su complejidad estructural (Axiom).

Características:

- Presenta una abundante y calificada documentación, con una amplia comunidad que lo soporta.
- Presenta una interfaz de usuario por defecto en modo texto.

- Disponible en distintos sistemas operativos: GNU/Linux, Mac OSX, MS Windows.
- Soporte de datos estructurados, lenguaje de programación que impulsa la programación estructurada. (autores, 2003)

1.4.5 Yacas

Yacas: *Yet Another Computer Algebra System*, es un entorno de cálculo simbólico de propósito general. Tiene entre sus funcionalidades: cálculo en precisión arbitraria, aritmética racional, números complejos, cálculo de derivadas, resolución de ecuaciones (simbólica y numéricamente), entre otras (Yacas).

Características:

- Es un programa moderno, escrito en C++, que es desarrollado por una comunidad muy dinámica.
- Presenta una amplia documentación.
- Existen versiones para numerosos sistemas operativos: GNU/Linux, Mac OSX, UNIX y MS Windows.
- Para interactuar con el usuario se utiliza la clásica consola de texto donde se introducen los comandos de Yacas.
- Permite hacer gráficos en 2D y 3D.

1.4.6 GNU Octave

Sistema desarrollado en C++, con interfaz a modo de comandos especializado en el procesamiento numérico. Cuenta con un lenguaje compatible con *Matlab* (considerado su equivalente comercial).

Entre varias características que comparten se puede destacar que ambos ofrecen un intérprete permitiendo ejecutar órdenes en modo interactivo. *Octave* no es un sistema de álgebra computacional como podría ser *Maxima*, sino que usa un lenguaje que está orientado al análisis numérico (Eaton, 2008).

Características:

- *Octave* está escrito en C++ usando la librería STL.
- Tiene un intérprete de su propio lenguaje y permite una ejecución interactiva o por lotes.

- Puede extenderse el lenguaje con funciones y procedimientos por medios de módulos dinámicos.
- Utiliza otros programas GNU para ofrecer al usuario crear gráficos para luego imprimirlos o guardarlos.
- Octave también se comporta como una consola de órdenes. Permite listar contenidos de directorios.
- Corre en plataformas Unix y Windows.
- Puede cargar archivos con funciones de *Matlab* de extensión *.m*.
- Cuenta con un lenguaje *Octave* propio interpretado, el cual soporta gran parte de las funciones de la librería estándar de C. Está pensado para trabajar con matrices y provee mucha funcionalidad para ello.
- Puede extenderse para ofrecer compatibilidad a las llamadas al sistema *UNIX*.

1.4.7 Euler

Es un programa de cálculo numérico matricial. Contiene, tanto en sus versiones GNU/Linux como MS Windows una interfaz de usuario cómoda y buenos gráficos en 2D y 3D (Euler sourceforge).

Características:

- Tiene un moderno lenguaje de programación.
- Incluye las funciones básicas requeridas para este tipo de programas: integración y diferenciación numérica, estadística, interpolación, transformada de *Fourier*, ecuaciones diferenciales y otras.

1.4.8 R.

Herramienta matemática expresada como un entorno y un lenguaje de programación orientado a objetos de cálculo, fundamentalmente estadístico, y generación de gráficos (Yurramendi, 2007 pág. 7). R es un lenguaje interpretado, con una sintaxis simple e intuitiva.

R se distribuye gratuitamente bajo los términos de la GNU General Public Licence; su desarrollo y distribución son llevados a cabo por varios estadísticos conocidos como el Grupo Nuclear de Desarrollo de R (Paradis, 2002).

Se encuentra disponible en plataformas como: Windows, Linux (Debian, Mandrake, RedHat, SuSe), Macintosh y Alpha Unix.

1.5 Herramienta libre seleccionada para desarrollar la propuesta.

La herramienta libre seleccionada para desarrollar la propuesta en esta investigación ha sido Maxima.

Justificación de la selección:

Maxima es un sistema con licencia libre que posee funcionalidades para cubrir los contenidos del plan de estudio de la asignatura Matemática III de la disciplina Matemática General de la carrera Ingeniería Informática. Tiene un procesamiento fundamentalmente simbólico, lo que posibilita la obtención de modelos matemáticos. Cuenta con funciones para la solución analítica de EDO, de sistema de EDO lineales, para la representación de funciones mediante series de potencias (series de *Taylor*) y series de *Fourier*.

Cuenta con un lenguaje de programación que permite a los estudiantes de informática desarrollar sus propios algoritmos y de esta forma contribuir a la formación del futuro profesional.

Maxima cubre las necesidades básicas de cálculo simbólico para un alumno de primer curso de una titulación universitaria (Galván, 2005), (Galván, 2007), (Pardini, 2007) y sirve de herramienta investigativa en proyectos reales.

Es por esto que Maxima ha sido de todas las herramientas libres estudiadas y expuestas anteriormente, la que mejor se adecua para el desarrollo de la herramienta propuesta en la investigación.

1.6 Metodología de desarrollo.

Las metodologías de desarrollo pueden clasificarse en Metodologías Estructuradas o Metodologías Orientadas a Objetos, de acuerdo a las notaciones utilizadas para especificar artefactos producidos en actividades de análisis y diseño; y en Metodologías Tradicionales o Metodologías Ágiles si se considera su filosofía de desarrollo.

1.6.1 Metodologías Estructuradas y Metodologías Orientadas a Objetos.

Las Metodologías Estructuradas se basan en la descomposición funcional de problemas en unidades más pequeñas interrelacionadas entre sí. Representan los procesos, flujos y estructuras de datos, de una manera jerárquica y ven el sistema como entradas-proceso-salidas. Estas metodologías funcionan muy bien con los lenguajes de programación estructurados.

El principio fundamental del enfoque estructurado es examinar al sistema desde las funciones y tareas, mientras que las Metodologías Orientadas a Objetos modelan el sistema examinando el dominio del problema como un conjunto de objetos que interactúan entre sí.

Con la aparición del paradigma de la orientación a objetos (OO) surgieron métodos, procesos y metodologías específicas como OMT, Objectory, RUP o Métrica 3 (en su enfoque OO), entre otras. En estas metodologías priman las fases, actividades y tareas antes que las personas: lo más importante es el rol que juega la persona dentro del proceso de desarrollo.

1.6.2 Metodologías Tradicionales y Metodologías Ágiles.

Las metodologías con mayor énfasis en la planificación y control del proyecto se denominan Metodologías Tradicionales o Pesadas (Letelier). Imponen una disciplina de trabajo sobre el proceso de desarrollo del software, con el fin de conseguir un software más eficiente. Para ello, se hace énfasis en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del producto software. Se centran en el control del proceso mediante una rigurosa definición de roles, actividades, artefactos, herramientas y notaciones para el modelado y la documentación detallada (Daniele, 2007).

Las Metodologías Ágiles trabajan con requisitos desconocidos o variables. Estas se utilizan si no existen requisitos estables y si no existe una gran posibilidad de tener un diseño estable y de seguir un proceso totalmente planificado, que no vaya a variar ni en tiempo, ni en dinero. Estas metodologías facilitan la generación rápida de prototipos y de versiones previos a la entrega final, lo cual favorece al cliente.

1.6.3 Metodología seleccionada para desarrollar la propuesta.

Debido a la característica de la herramienta propuesta que parte de la concepción de un sistema estructurado, la metodología seleccionada para desarrollar la misma es la conocida como: métodos de análisis y diseño estructurado difundidos por Yourdon, Weinberg y De Marco.

Esta metodología utilizada con frecuencia en las décadas de 1970 y 1980 se destaca por su claridad, sencillez, declaración simple de artefactos, diagramas fáciles de manipular y reducido número de símbolos para la representación de diagramas. Mantiene la filosofía jerárquica propia de las metodologías para sistemas estructurados y consta de cuatro herramientas que permiten el adecuado uso de la misma.

1.7 Conclusiones

En el presente capítulo, luego de realizar un estudio teórico y justificar el desarrollo de una herramienta para apoyar el PDE de la asignatura Matemática III y analizar herramientas libres disponibles para el desarrollo de la investigación, se ha fundamentado la selección del sistema Maxima para desarrollar la propuesta y de la metodología análisis y diseño estructurado difundidos por Yourdon, Weinberg y De Marco como metodología a utilizar.

Capítulo II. Descripción y construcción de la solución propuesta.

2.1 Introducción.

En este capítulo se presenta la concepción general del sistema propuesto así como los requerimientos mínimos de rendimiento, de software y de hardware para el uso del mismo. Se utiliza una metodología que integra métodos de análisis y diseño de sistemas estructurados, en la que se detallan: diagramas de flujo de datos, la representación lógica de los procesos computacionales (algoritmos), el diccionario de datos y se describe la utilización integral de todos estos elementos.

2.2 Concepción general del sistema.

Maxima, asistente matemático seleccionado para desarrollar la presente propuesta, tiene tres interfaces de usuario. La interfaz utilizada para el desarrollo de la investigación fue xMaxima, la que se caracteriza por su ambiente a modo de comandos. Consta de dos paneles principales en su ventana principal: la parte superior donde se introducen los comandos y se obtienen los resultados, y la parte inferior en la que por defecto aparece una página web con elementos básicos fundamentales para el trabajo con Maxima (Ver Anexo 1).

El resultado a alcanzar con esta investigación es el desarrollo de una herramienta informática basada en el sistema Maxima para apoyar el PDE de la asignatura Matemática III perteneciente a la Disciplina Matemática General en la carrera Ingeniería Informática. Para lograr este propósito se debe crear una página web principal que tenga todos los contenidos de la asignatura organizados por temas y que el estudiante pueda cargar en el navegador de xMaxima. Por otra parte existen contenidos de la asignatura para los cuales xMaxima no tiene comandos creados y, por tanto, deben ser programados utilizando el lenguaje de Maxima y guardados como un fichero .mac para que el estudiante los pueda cargar y utilizar.

La página web desarrollada debe ser cargada en el navegador de xMaxima (ubicado en la parte inferior de la ventana principal) de forma tal que se mantenga la misma dirección de la página que se muestra por defecto y se cambie solamente el nombre de esta página por `matematicaIII.html`. Para lograr esto la página desarrollada (`matematicaIII.html`) debe ser copiada en la misma dirección que la página que muestra xMaxima por defecto y que se ubica en la instalación del software.

La página web `matematicaIII.html` debe estar estructurada en dos temas fundamentales (Ver Anexo 2):

- Ecuaciones diferenciales ordinarias: En este tema la página web debe contemplar instrucciones para resolver de forma analítica ecuaciones diferenciales ordinarias (EDO) de primer y segundo orden, lineales de orden superior y sistema de EDO lineales. Solución general y particular de EDO. Métodos por los que son resueltas las EDO, así como problemas de valores iniciales y de frontera.
- Series: En este tema la página web debe contemplar instrucciones para determinar convergencia o divergencia de series numéricas, desarrollo de funciones en series de Taylor (de potencias) y de Fourier.

En la página web debe quedar bien explicada la forma de pasar cada parámetro a las distintas instrucciones, además de tener vínculos hacia otras dos páginas web donde deben aparecer ejemplos en una y el código fuente de las instrucciones programadas en la otra.

El fichero de extensión `.mac`, que llevará el título `matematicaIII.mac`, debe contener aquellas funcionalidades que Maxima no posee para abarcar todo el contenido de la asignatura Matemática III y debe ser ubicado dentro de la carpeta de la instalación de Maxima en el mismo lugar donde se encuentren los archivos `.mac`. A continuación se muestran funciones (que deben ser programadas) correspondientes a estas funcionalidades:

- `separable(p, q, x, y)`: Para obtener soluciones de EDO de variables separables.
- `homogenea(r, x, y)`: Para obtener soluciones de EDO homogéneas.
- `exacta(p, q, x, y)`: Para obtener soluciones de EDO exactas.

- $\text{factor_integrante}(p, q, x, y)$: Para obtener un factor integrante en términos de una sola variable que convierta una EDO en exacta.
- $\text{reducible_exacta}(p, q, x, y)$: Para obtener soluciones de EDO reducibles a exactas.
- $\text{lineal_o1}(p, q, x, y)$: Para obtener soluciones de EDO lineales.
- $\text{bernoulli}(p, q, k, x, y)$: Para obtener soluciones de EDO reducibles a lineales o de Bernoulli.
- $\text{termino_enesimo}(a_n, n, n_i)$: Para evaluar el criterio del término enésimo en series numéricas.
- $\text{comparacion}(a_n, b_n, n)$: Para evaluar el criterio de comparación, variante paso al límite en series numéricas de términos no negativos.
- $\text{raiz}(a_n, n, n_i)$: Para evaluar el criterio de la raíz en series numéricas de términos no negativos.
- $\text{cociente}(a_n, n, n_i)$: Para evaluar el criterio del cociente en series numéricas de términos no negativos.
- $\text{raabe}(a_n, n, n_i)$: Para evaluar el criterio de Raabe en series numéricas de términos no negativos.
- $\text{gauss}(a_n, n, n_i)$: Para evaluar el criterio de Gauss en series numéricas de términos no negativos.
- $\text{comp_serie}(a_n, n, n_i)$: Para evaluar la convergencia por cualquier criterio en series numéricas de términos no negativos.

2.3 Requerimientos mínimos del sistema.

Requerimientos de rendimiento.

Los tiempos de respuesta deben depender de las características de los datos que se estén procesando.

Se debe garantizar un alto nivel de confiabilidad y precisión en la respuesta.

Requerimientos de soporte.

El sistema debe posibilitar su perfeccionamiento y la incorporación de nuevas opciones en un futuro.

Requerimientos de ayuda y documentación en línea.

El sistema debe disponer de una ayuda bien detallada sobre las principales opciones que brinda.

La herramienta debe contar con una documentación básica que comprenda los aspectos generales a tener en cuenta para utilizar eficientemente los requerimientos.

Requerimiento de software.

Para la instalación del sistema se debe disponer de cualquier versión del sistema operativo GNU/Linux o Microsoft Windows.

Requerimiento de hardware.

Para la instalación del sistema se requiere:

Procesador de velocidad 1GHz.

- 128 MB de RAM
- 1 GB de HDD Libre

2.4 Análisis y diseño estructurado.

2.4.1 Principios básicos del análisis y diseño estructurado.

El análisis y diseño estructurado de un sistema informático se lleva a cabo partiendo de lo general hacia lo particular. En este capítulo se utilizarán los métodos de análisis y diseño estructurado difundidos por Yourdon, Weinberg y De Marco que siguen este principio.

La esencia de esta metodología consiste en definir un sistema general correspondiente al mayor nivel de agregación posible, el cual se representa con un solo diagrama. Luego este nivel se descompone en sistemas más detallados, los cuales se representan por varios diagramas, y así se va desglosando hasta llegar a un nivel que no admite más descomposición.

En este lenguaje de modelación se emplea un reducido número de símbolos para representar todo el flujo informativo que se produce en el sistema, lo cual hace sencillo y fácil de manipular este lenguaje. La notación que se utiliza es la siguiente: saetas para indicar el flujo de datos entre procesos, círculos para indicar los procesos y cuadrados para indicar fuentes o destinos de información.

Las herramientas que forman parte de esta metodología de análisis y diseño estructurado son las siguientes:

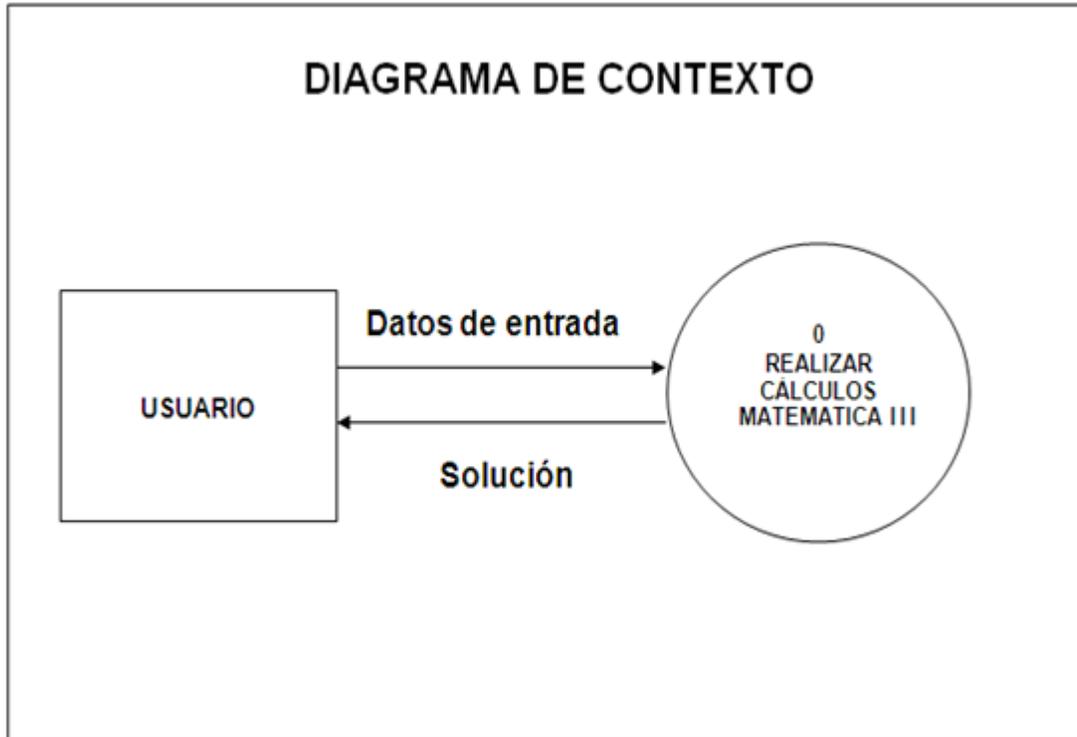
- Diagramas de flujo de datos (DFD).
- Representación lógica en los procesos computacionales. Lenguaje natural estructurado (LNE).
- Diccionario de datos (DD).
- Diagrama de estructura de datos (DED).

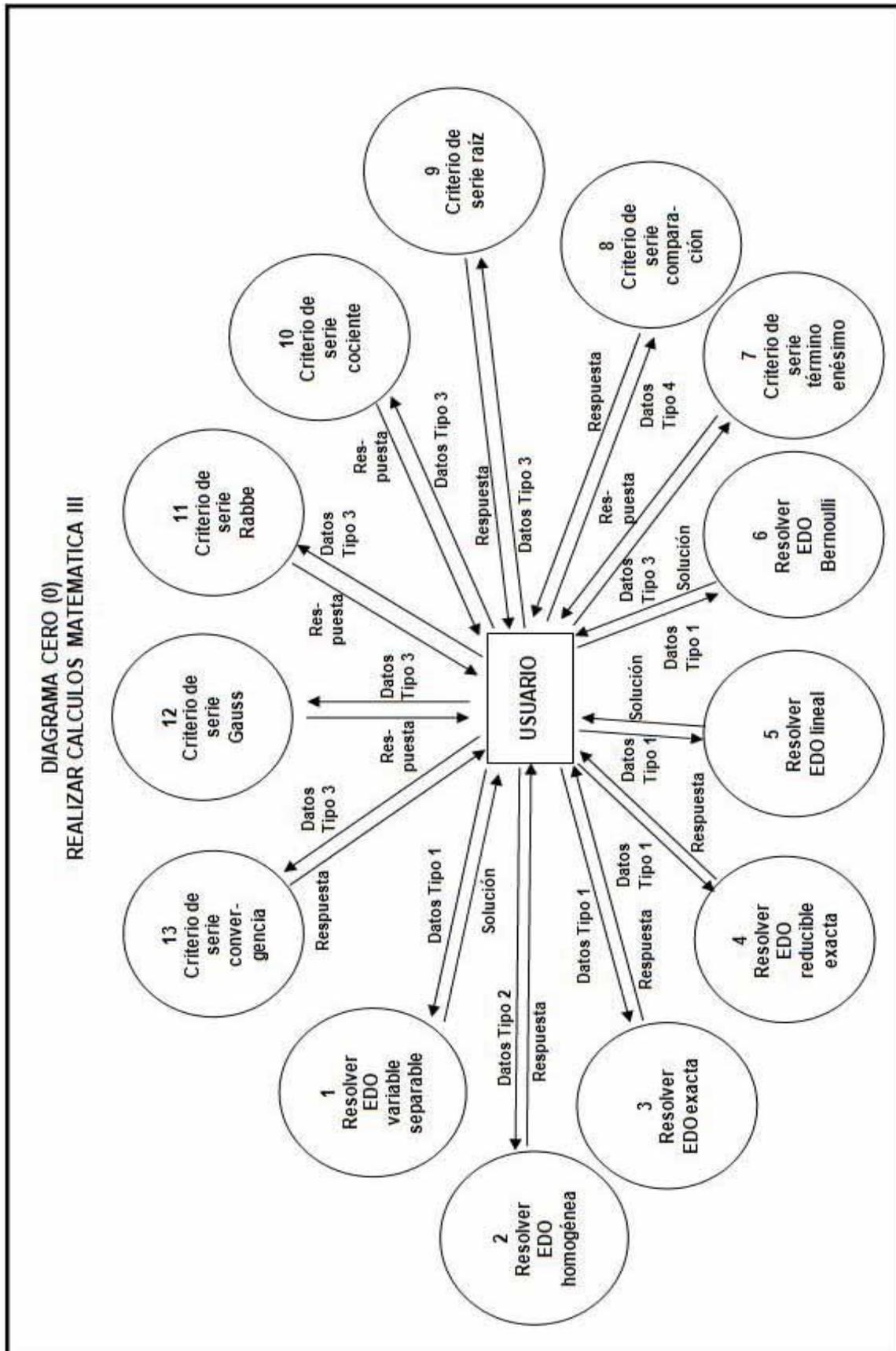
Estas herramientas juntas forman un conjunto integrado. A continuación se presentan cada una de ellas a excepción del diagrama de estructura de datos (DED) porque en la herramienta desarrollada no se emplean ficheros para el almacenamiento de información ni bases de datos.

2.4.2 Diagramas de flujo de datos (DFD).

Los diagramas de flujo de datos (DFD) permiten representar sistemas, definiendo a estos en términos de sus componentes y de las relaciones informativas existentes entre ellos.

Los DFD se elaboran partiendo de lo más general (en este caso se llama “Diagrama de contexto”), bajando a un nivel más detallado (“Diagrama cero”) y llegando hasta los niveles más elementales, donde el proceso ya no admite más descomposición, con lo que se llega a las “Tareas funcionales” (Blanco Encinosa, y otros, 1991 pág. 208).





2.4.3 Representación lógica en los procesos computacionales. Lenguaje natural estructurado (LNE).

Los procesos elementales que se producen conocidos como tareas funcionales, se describen utilizando herramientas que puedan detallarlos perfectamente. En la investigación se emplea la herramienta lenguaje natural estructurado (LNE) para este fin.

El lenguaje estructurado es una adaptación del lenguaje natural y un acercamiento al lenguaje computacional (lenguaje de programación), utilizado para describir los procesos lógicos que transcurren en una tarea. No es propiamente un lenguaje de programación, sino un método de plantear un problema de forma tal que resulte sencilla y simple su comprensión y su traducción a un lenguaje de programación convencional (Blanco Encinosa, y otros, 1991 pág. 245).

A continuación se muestran en LNE las nuevas tareas implementadas:

función solución EDO variables separable(p, q, x, y)

hacer $I_p = \int p \, dx$

hacer $I_q = \int q \, dy$.

devolver $I_q = I_p$

Fin solución EDO variables separable.

función solución EDO homogénea(r, x, y)

Si r es homogénea **Entonces**

hacer $y = ux$.

poner r en función de u y x

resolver por variables separables con el nuevo cambio de variable.

terminar

Sino

mostrar mensaje de que la EDO no es homogénea.

Fin Si

Fin solución EDO homogénea.

función solución EDO exacta(p, q, x, y)

$$\text{hacer } Dqx = \frac{dq}{dx}$$

$$\text{hacer } Dpy = \frac{dp}{dy}$$

Si $Dqx = Dpy$ **Entonces**

$$\text{hacer } Ipx = \int p \, dx$$

$$\text{hacer } D(Ipx)y = \frac{dIpx}{dy}.$$

$$\text{hacer } Iy = \int (q - D(Ipx)y) \, dy.$$

$$\text{devolver } Ipx + Iy.$$

terminar

Sino

mostrar mensaje de que la EDO no es exacta.

Fin Si

Fin solución EDO exacta.

función solución EDO reducible exacta(p, q, x, y)

FI = calcular factor integrante

Si FI se encontró **Entonces**

multiplicar p y q por el factor integrante (FI)

calcular por el método de exacta

terminar

Sino

mostrar mensaje de que la EDO no es reducible a exacta.

Fin Si

Fin solución EDO reducible exacta.

función solución EDO lineal(p, q, x, y)

hacer $Ipx = \int p \, dx$

hacer $F = e^{Ipx}$

devolver $\int (q * F) dx / F$.

Fin solución EDO lineal.

función solución EDO bernoulli(p, q, x, y)

hacer $Ipx = \int p \, dx$

hacer $F = e^{(1-k)*Ipx}$

devolver $y^{(1-k)} = (1 - k) * \int (q * F) dx / F$.

Fin solución EDO bernoulli.

función criterio termino_enesimo(an, n, ni)

hacer $Tn = \lim_{n \rightarrow \infty} an$.

Si $Tn \neq 0$ **Entonces**

devolver que $\sum_{n=ni}^{\infty} an$ diverge.

Sino

mostrar mensaje que el criterio no decide.

Fin Si

Fin criterio termino_enesimo.

función criterio comparacion(an, bn, n)

hacer $C = \lim_{n \rightarrow \infty} an/bn$.

Si $C > 0$ Entonces

mostrar mensaje $\sum_n^\infty an$ y $\sum_n^\infty bn$ tienen el mismo comportamiento.

Si $C = 0$

de la convergencia de $\sum_n^\infty bn$ se deduce la convergencia de $\sum_n^\infty an$ y de la divergencia de $\sum_n^\infty an$ se deduce la divergencia de $\sum_n^\infty bn$.

Sino

mostrar mensaje que el criterio no decide.

Fin Si

Fin criterio comparacion.

función criterio raiz(an, n, ni)

hacer $R = \lim_{n \rightarrow \infty} an^{1/n}$.

Si $R < 1$ Entonces

devolver que $\sum_{n=ni}^\infty an$ converge.

Si $R > 1$ Entonces

devolver que $\sum_{n=ni}^\infty an$ diverge.

Sino

mostrar mensaje que el criterio no decide.

Fin Si

Fin criterio raíz.

función criterio cociente(an, n, ni)

$$\text{hacer } C = \lim_{n \rightarrow \infty} \frac{a_{n+1}}{a_n}.$$

Si $C < 1$ Entonces

devolver que $\sum_{n=ni}^{\infty} an$ converge.

Si $C > 1$ Entonces

devolver que $\sum_{n=ni}^{\infty} an$ diverge.

Sino

mostrar mensaje que el criterio no decide.

Fin Si

Fin criterio cociente.

función criterio raabe(an, n, ni)

$$\text{hacer } Ra = \lim_{n \rightarrow \infty} n * \left(1 - \frac{a_{n+1}}{a_n}\right).$$

Si $Ra > 1$ Entonces

devolver que $\sum_{n=ni}^{\infty} an$ converge.

Si $Ra < 1$ Entonces

devolver que $\sum_{n=ni}^{\infty} an$ diverge.

Sino

mostrar mensaje que el criterio no decide.

Fin Si

Fin criterio raabe.

función criterio gauss(an, n, ni)

representar el cociente $\frac{a_{n+1}}{a_n} = 1 - \frac{L}{n} + \frac{Cn}{n^2}$.

Si Cn está acotada superiormente y $L > 1$ **Entonces**

devolver que $\sum_{n=ni}^{\infty} an$ converge.

Sino

devolver que $\sum_{n=ni}^{\infty} an$ diverge.

Fin Si

Fin criterio gauss.

función comp_serie(an, n, ni)

Si el criterio del término enésimo decide **Entonces**

devolver que $\sum_{n=ni}^{\infty} an$ diverge.

Si el criterio de comparación decide **Entonces**

devolver el resultado del criterio.

Si el criterio de la raíz decide **Entonces**

devolver el resultado del criterio.

Si el criterio del cociente decide **Entonces**

devolver el resultado del criterio.

Si el criterio de Raabe decide **Entonces**

devolver el resultado del criterio.

Si el criterio de Gauss decide **Entonces**

devolver el resultado del criterio.

Sino

mostrar mensaje que no se tiene criterio para el análisis de la convergencia de la serie.

Fin Si

Fin criterio comp_serie.

2.4.4 Diccionario de datos (DD).

El diccionario de datos es una herramienta que permite definir con mayor claridad los términos asociados con el desarrollo de un sistema. Ayuda a eliminar ambigüedades, indefiniciones, definiciones vagas o no precisas. Permite definir los datos, los procesos lógicos y físicos que transcurren en el sistema, tanto desde el punto de vista del usuario (proceso lógico) como del programador (proceso físico) (Blanco Encinosa, y otros, 1991).

El DD consta de tres definiciones fundamentales:

- Definición de flujos de datos (Representados en el Diagrama cero (0) con saetas): Especifica los flujos que van desde el usuario hasta las tareas y que devuelven las tareas hacia el usuario.
- Definición de elementos de datos: Especifica datos que manipulan y procesan las tareas. Generalmente son los argumentos que reciben las tareas.
- Definición de tareas funcionales: Especifica los procesos terminales representados en el Diagrama cero (0) con círculos.

Definición de flujo de datos	
Nombre del flujo de datos	Datos Tipo 1
Esquema	
<p>El diagrama muestra un usuario en un recuadro rectangular a la derecha que envía un flujo de datos etiquetado como 'Datos Tipo 1' a un proceso circular a la izquierda. El proceso circular contiene el texto: '1 Resolver EDO variable separable'.</p>	
Descripción	Datos que se utilizan para resolver EDO por métodos analíticos. Está compuesto por una función p que depende solamente de x , una función q que depende solamente de y si se utiliza en los

	procesos 1, 5 y 6. En los restantes procesos que se utiliza este tipo de flujo de dato la función p y q dependen de x y y respectivamente. Además, está compuesto por x que representa la variable independiente y y que representa la variable dependiente.
Composición	función p , función q , variable independiente x , variable dependiente y .
Fuente	Usuario
Destino	Procesos: 1- Resolver EDO por variables separables. 3- Resolver EDO exactas. 4- Resolver EDO reducible a exactas. 5- Resolver EDO lineales. 6- Resolver EDO reducible a lineales (Bernoulli).

Tabla 1. Definición de flujo de datos. Datos tipo 1.

Definición de flujo de datos	
Nombre del flujo de datos	Datos Tipo 2
Esquema	<pre> graph LR USUARIO[USUARIO] -- Datos Tipo 2 --> PROCESO((2 Resolver EDO homogénea)) </pre>
Descripción	Datos que se utilizan para resolver EDO por métodos analíticos. Está compuesto por una función r que dependen de x y y respectivamente, por x que representa la variable independiente y y que representa la variable dependiente.
Composición	función r , variable independiente x , variable dependiente y .

Fuente	Usuario
Destino	Proceso: 2- Resolver EDO homogéneas.

Tabla 2. Definición de flujo de datos. Datos tipo 2.

Definición de flujo de datos	
Nombre del flujo de datos	Datos Tipo 3
<p>Esquema</p> <pre> graph LR USUARIO[USUARIO] -- Datos Tipo 3 --> Criterio((9 Criterio de serie raíz)) </pre>	
Descripción	Datos que se utilizan para evaluar criterios de convergencia de series numéricas. Está compuesto por el término enésimo an de una serie numérica, la variable de la serie n , y el valor numérico donde comienza la series numérica ni .
Composición	Término enésimo an , variable de la serie n , valor en que comienza la serie ni .
Fuente	Usuario
Destino	Proceso: 7- Criterio de serie término enésimo. 9- Criterio de serie raíz. 10- Criterio de serie cociente. 11- Criterio de serie Raabe. 12- Criterio de serie Gauss. 13- Criterio de serie convergencia general.

Tabla 3. Definición de flujo de datos. Datos tipo 3.

Definición de flujo de datos	
Nombre del flujo de datos	Datos Tipo 4

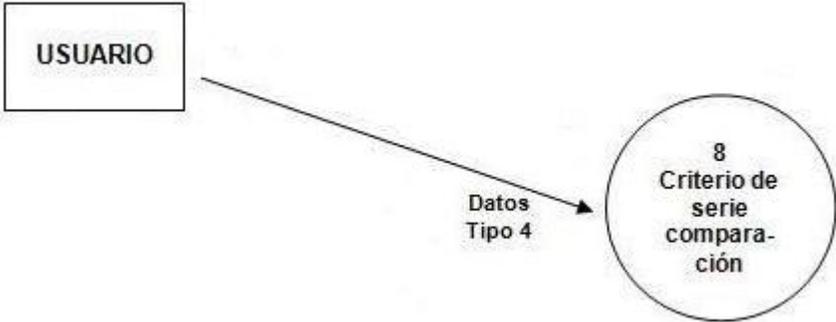
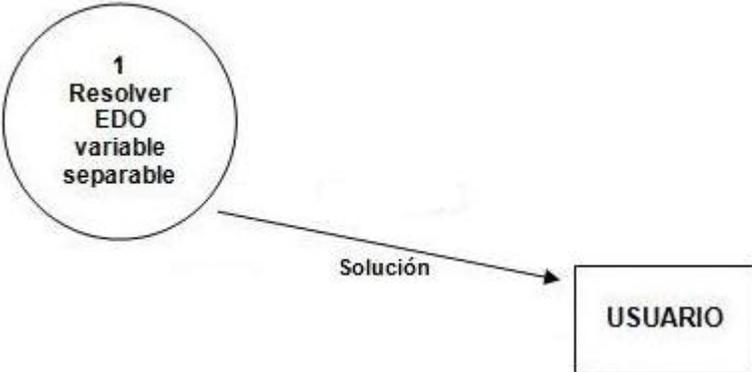
<p>Esquema</p> 	
Descripción	Datos que se utilizan para evaluar el criterio de convergencia de series numéricas: comparación paso al límite. Está compuesto por el término enésimo an de una serie numérica de la que se quiere evaluar su convergencia, el término enésimo bn de una serie numérica que se conoce su comportamiento (si diverge o converge) y la variable de la serie n .
Composición	término enésimo an , término enésimo bn , variable de la serie n .
Fuente	Usuario
Destino	Proceso: 8- Criterio de serie término enésimo.

Tabla 4. Definición de flujo de datos. Datos tipo 4.

Definición de flujo de datos	
Nombre del flujo de datos	Solución.
<p>Esquema</p> 	
Descripción	La Solución se emite al usuario como resultado de la ejecución de los procesos

	1, 5 y 6. Está conformada por modelos matemáticos que se obtienen como resultado de los cálculos correspondientes.
Composición	Compuesto por una expresión (modelo) simbólica matemática.
Fuente	Procesos: 1- Resolver EDO por variables separables. 5- Resolver EDO lineales. 6- Resolver EDO reducible a lineales
Destino	Usuario.

Tabla 5. Definición de flujo de datos. Solución.

Definición de flujo de datos	
Nombre del flujo de datos	Respuesta.
Esquema	<pre> graph LR A((2 Resolver EDO homogénea)) -- Respuesta --> B[USUARIO] </pre>
Descripción	La respuesta se emite al usuario como resultado de la ejecución de los procesos: 2, 3, 4, 7, 8, 9, 10, 11, 12 y 13. Está conformada por modelos matemáticos que se obtienen como resultado de los cálculos correspondientes o por mensajes al usuario.
Composición	Compuesto por una expresión (modelo) simbólica matemática o mensajes de respuesta al usuario.
Fuente	Procesos: 2- Resolver EDO homogéneas. 3- Resolver EDO exactas. 4- Resolver EDO reducible a exactas. 7- Criterio de serie término enésimo. 8- Criterio de serie comparación. 9- Criterio de serie raíz. 10- Criterio de serie cociente. 11- Criterio de serie Raabe. 12- Criterio de serie Gauss.

	13- Criterio de serie convergencia general.
Destino	Usuario.

Tabla 6. Definición de flujo de datos. Respuesta.

Definición del elemento de datos	
Nombre del flujo de datos	Función.
Descripción.	Expresión analítica de una función real de una variable independiente o dos variables reales.
Composición.	Elementos que se pueden involucrar en la expresión analítica de una función real de una variable real o de dos variables reales.
Procesos asociados.	1- Resolver EDO por variables separables. 2- Resolver EDO homogéneas. 3- Resolver EDO exactas. 4- Resolver EDO reducible a exactas. 5- Resolver EDO lineales. 6- Resolver EDO reducible a lineales (Bernoulli).
Características del dato.	Tipo cadena

Tabla 7. Definición del elemento de datos. Función.

Definición del elemento de datos	
Nombre del flujo de datos	Variable.
Descripción.	Por defecto se utiliza la letra x que representa la variable independiente y la letra y que representa la variable dependiente de algunas de las funciones de los procesos.
Composición.	Combinación de letras.
Procesos asociados.	1- Resolver EDO por variables separables. 2- Resolver EDO homogéneas. 3- Resolver EDO exactas. 4- Resolver EDO reducible a exactas. 5- Resolver EDO lineales. 6- Resolver EDO reducible a lineales (Bernoulli).
Características del dato.	Tipo cadena

Tabla 8. Definición del elemento de datos. Variable.

Definición del elemento de datos	
Nombre del flujo de datos	Término enésimo.
Descripción.	Expresión .analítica de la forma general de una serie numérica.
Composición.	Elementos que se pueden involucrar en la expresión analítica.
Procesos asociados.	7- Criterio de serie término enésimo. 8- Criterio de serie comparación. 9- Criterio de serie raíz. 10- Criterio de serie cociente. 11- Criterio de serie Raabe. 12- Criterio de serie Gauss. 13- Criterio de serie convergencia general.
Características del dato.	Tipo cadena

Tabla 9. Definición del elemento de datos. Término enésimo.

Definición del elemento de datos	
Nombre del flujo de datos	Variable de la serie.
Descripción.	Por defecto se utiliza la letra n que representa la variable en la cual está expresada la serie numérica.
Composición.	Combinación de letras.
Procesos asociados.	7- Criterio de serie término enésimo. 8- Criterio de serie comparación. 9- Criterio de serie raíz. 10- Criterio de serie cociente. 11- Criterio de serie Raabe. 12- Criterio de serie Gauss. 13- Criterio de serie convergencia general.
Características del dato.	Tipo cadena

Tabla 10. Definición del elemento de datos. Variable de la serie.

Definición del elemento de datos	
Nombre del flujo de datos	Valor numérico donde comienza la serie.
Descripción.	Valor real donde comienza la serie numérica.
Composición.	Número real.

Procesos asociados.	7- Criterio de serie término enésimo. 8- Criterio de serie comparación. 9- Criterio de serie raíz. 10- Criterio de serie cociente. 11- Criterio de serie Raabe. 12- Criterio de serie Gauss. 13- Criterio de serie convergencia general.
Características del dato.	Tipo número

Tabla 11. Definición del elemento de datos. Valor numérico donde comienza la serie.

Definición de Tarea Funcional	
Nombre de la tarea.	Resolver EDO por variables separables.
Número de identificación del DFD.	DFD 0.
Esquema.	
Descripción.	El usuario pasa los parámetros: funciones p y q respectivamente, variable independiente y variable dependiente. La función calcula, utilizando funciones que presenta Maxima, la integral de p con respecto a x y la integral de q con respecto a y . Luego se devuelve el modelo solución
Flujos de datos de entrada:	Datos Tipo 1.
Flujos de datos de salida:	Solución.

Proceso lógico:	LNE Epígrafe 2.4.4: función solución EDO variables separable(p, q, x, y).
-----------------	---

Tabla 12. Definición de Tarea Funcional. Resolver EDO variables separables.

Definición de Tarea Funcional	
Nombre de la tarea.	Resolver EDO homogéneas.
Número de identificación del DFD.	DFD 0.
Esquema.	
Descripción.	El usuario pasa los parámetros: funciones r , variable independiente y y variable dependiente. Inicialmente se comprueba si la función r es homogénea de grado 0. Si la función es homogénea se introduce el cambio de variable $y=ux$ y se expresa r en el nuevo cambio de variable. Luego se aplica variable separable. Si la función no es homogénea se devuelve un mensaje.
Flujos de datos de entrada:	Datos Tipo 2.
Flujos de datos de salida:	Respuesta.
Proceso lógico:	LNE Epígrafe 2.4.4: función solución EDO homogénea(r , x , y).

Tabla 13. Definición de Tarea Funcional. Resolver EDO homogéneas.

Definición de Tarea Funcional	
Nombre de la tarea.	Resolver EDO exacta.
Número de identificación del DFD.	DFD 0.

<p>Esquema.</p>	
<p>Descripción.</p>	<p>El usuario pasa los parámetros: funciones p y q respectivamente, variable independiente y variable dependiente. Inicialmente se comprueba si la EDO es exacta. Si es exacta se hace uso de la función integral para devolver el modelo solución. Si la EDO no es exacta se devuelve un mensaje.</p>
<p>Flujos de datos de entrada:</p>	<p>Datos Tipo 1.</p>
<p>Flujos de datos de salida:</p>	<p>Respuesta.</p>
<p>Proceso lógico:</p>	<p>LNE Epígrafe 2.4.4: función solución EDO exacta(p, q, x, y).</p>

Tabla 14. Definición de Tarea Funcional. Resolver EDO exacta.

Definición de Tarea Funcional	
Nombre de la tarea.	Resolver EDO reducible a exacta.
Número de identificación del DFD.	DFD 0.
Esquema.	

Descripción.	El usuario pasa los parámetros: funciones p y q respectivamente, variable independiente y variable dependiente. Inicialmente se busca un factor integrante en términos de una sola variable. Si se encuentra un factor integrante en términos de una sola variable se multiplica la EDO por él y se resuelve por exacta. Si no se encuentra un factor integrante se devuelve un mensaje.
Flujos de datos de entrada:	Datos Tipo 1.
Flujos de datos de salida:	Respuesta.
Proceso lógico:	LNE Epígrafe 2.4.4: función solución EDO reducible a exacta(p, q, x, y).

Tabla 15. Definición de Tarea Funcional. Resolver EDO reducible a exacta.

Definición de Tarea Funcional	
Nombre de la tarea.	Resolver EDO lineal.
Número de identificación del DFD.	DFD 0.

Esquema.	
Descripción.	El usuario pasa los parámetros: funciones p y q respectivamente, variable independiente y variable dependiente. Luego se hace uso de la función integral para devolver el modelo solución.
Flujos de datos de entrada:	Datos Tipo 1.
Flujos de datos de salida:	Solución.
Proceso lógico:	LNE Epígrafe 2.4.4: función solución EDO lineal(p, q, x, y).

Tabla 16. Definición de Tarea Funcional. Resolver EDO lineal.

Definición de Tarea Funcional	
Nombre de la tarea.	Resolver EDO Bernoulli.
Número de identificación del DFD.	DFD 0.
Esquema.	

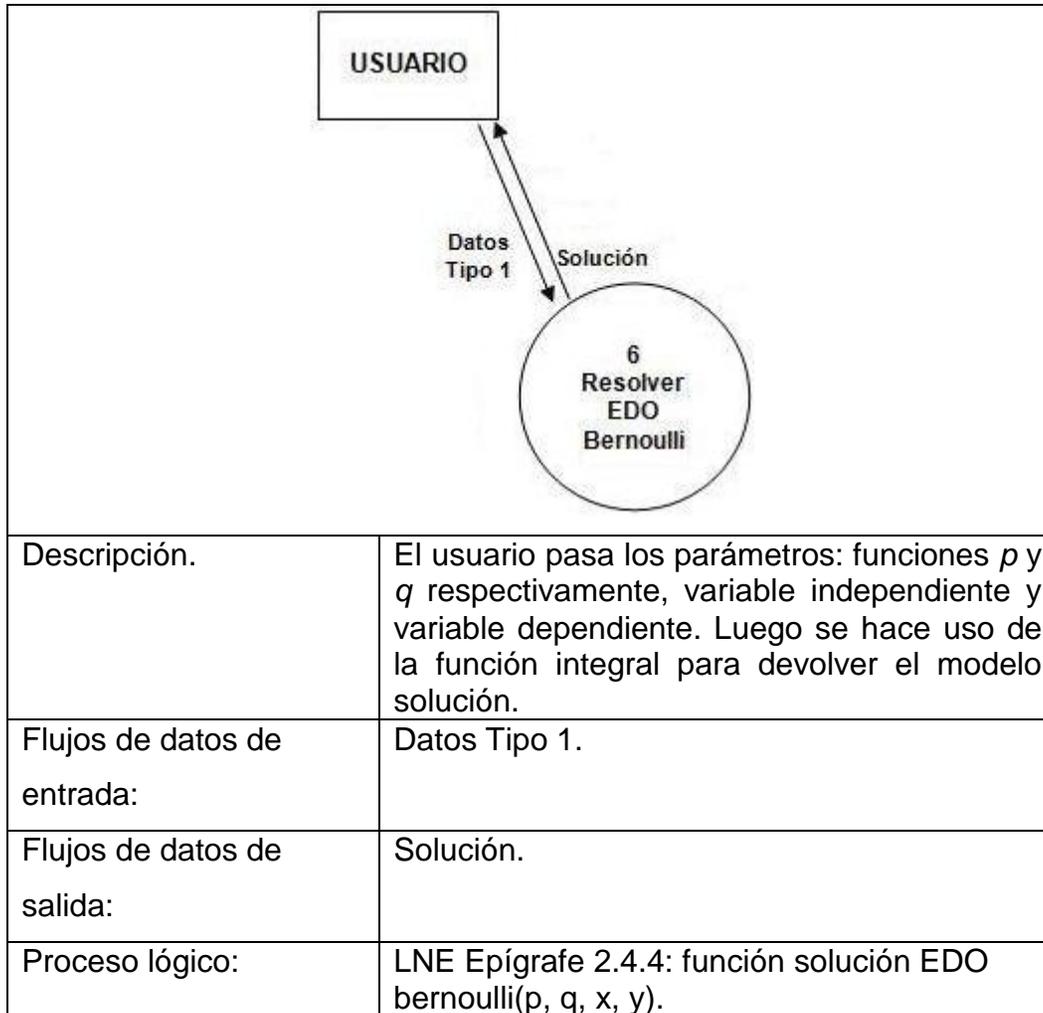


Tabla 17. Definición de Tarea Funcional. Resolver EDO Bernoulli.

Definición de Tarea Funcional	
Nombre de la tarea.	Criterio serie término enésimo.
Número de identificación del DFD.	DFD 0.
Esquema.	

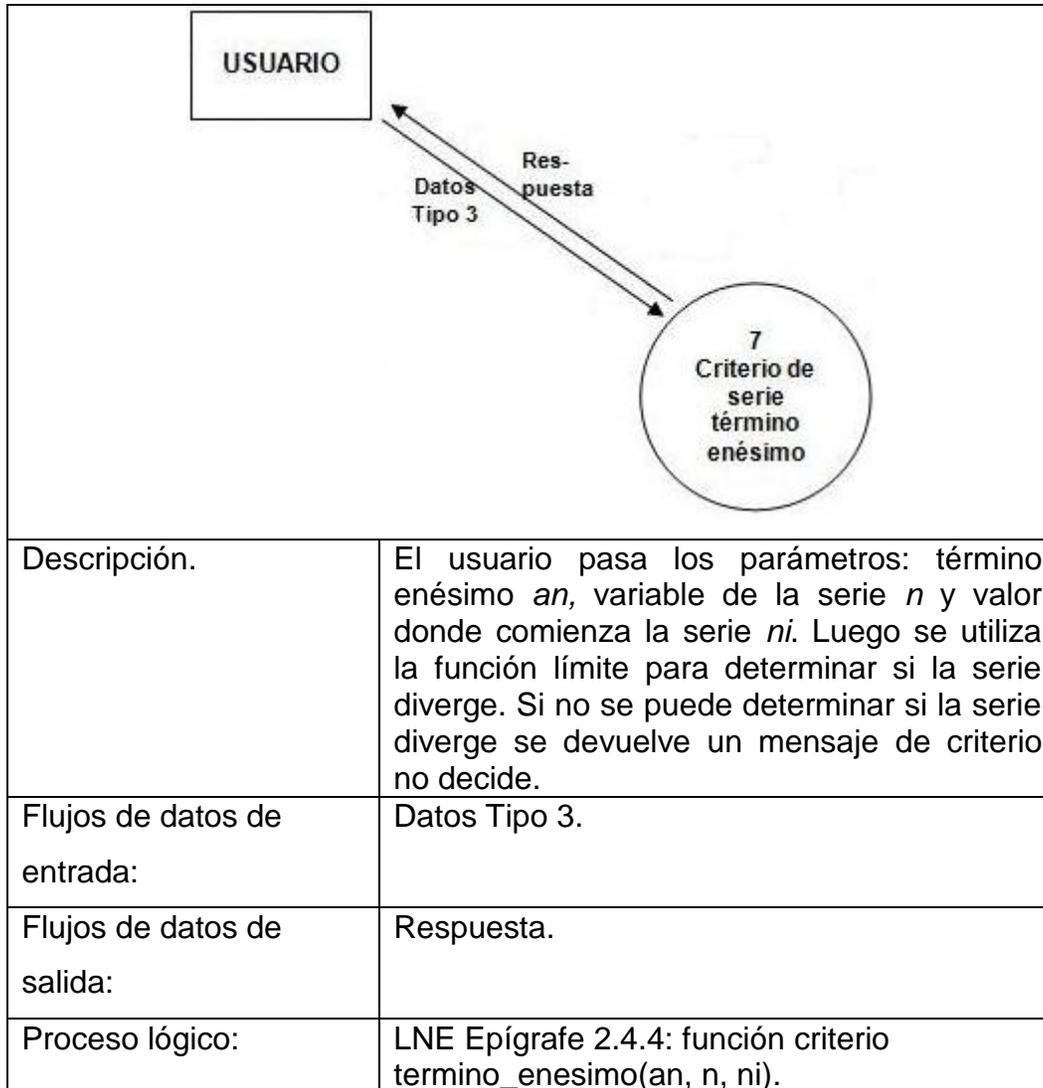


Tabla 18. Definición de Tarea Funcional. Criterio serie término enésimo.

Definición de Tarea Funcional	
Nombre de la tarea.	Criterio serie comparación.
Número de identificación del DFD.	DFD 0.

<p>Esquema.</p> <pre> graph LR U[USUARIO] -- "Datos Tipo 4" --> P((8 Criterio de serie comparación)) P -- "Respuesta" --> U </pre>	
<p>Descripción.</p>	<p>El usuario pasa los parámetros: término enésimo de la serie que se quiere analizar su convergencia a_n, término enésimo de la serie que se conoce su convergencia b_n y variable de la serie n. Luego se utiliza la función límite para determinar si las series tienen el mismo comportamiento o si de la convergencia/divergencia de una se deduce la convergencia/divergencia de la otra. Si no se puede determinar si la serie diverge o converge se devuelve un mensaje de criterio no decide.</p>
<p>Flujos de datos de entrada:</p>	<p>Datos Tipo 4.</p>
<p>Flujos de datos de salida:</p>	<p>Respuesta.</p>
<p>Proceso lógico:</p>	<p>LNE Epígrafe 2.4.4: función criterio comparacion(a_n, b_n, n).</p>

Tabla 19. Definición de Tarea Funcional. Criterio serie comparación.

Definición de Tarea Funcional	
Nombre de la tarea.	Criterio serie raíz.
Número de identificación del DFD.	DFD 0.

<p>Esquema.</p> <pre> graph LR Usuario[USUARIO] -- "Datos Tipo 3" --> Proceso((9 Criterio de serie raíz)) Proceso -- "Respuesta" --> Usuario </pre>	
<p>Descripción.</p>	<p>El usuario pasa los parámetros: término enésimo an, variable de la serie n y valor donde comienza la serie ni. Luego se utiliza la función raíz enésima y límite para determinar si la serie converge o diverge. Si no se puede determinar si la serie converge o diverge se devuelve un mensaje de criterio no decide.</p>
<p>Flujos de datos de entrada:</p>	<p>Datos Tipo 3.</p>
<p>Flujos de datos de salida:</p>	<p>Respuesta.</p>
<p>Proceso lógico:</p>	<p>LNE Epígrafe 2.4.4: función criterio raíz(an, n, ni).</p>

Tabla 20. Definición de Tarea Funcional. Criterio serie raíz.

Definición de Tarea Funcional	
Nombre de la tarea.	Criterio serie cociente.
Número de identificación del DFD.	DFD 0.
Esquema.	

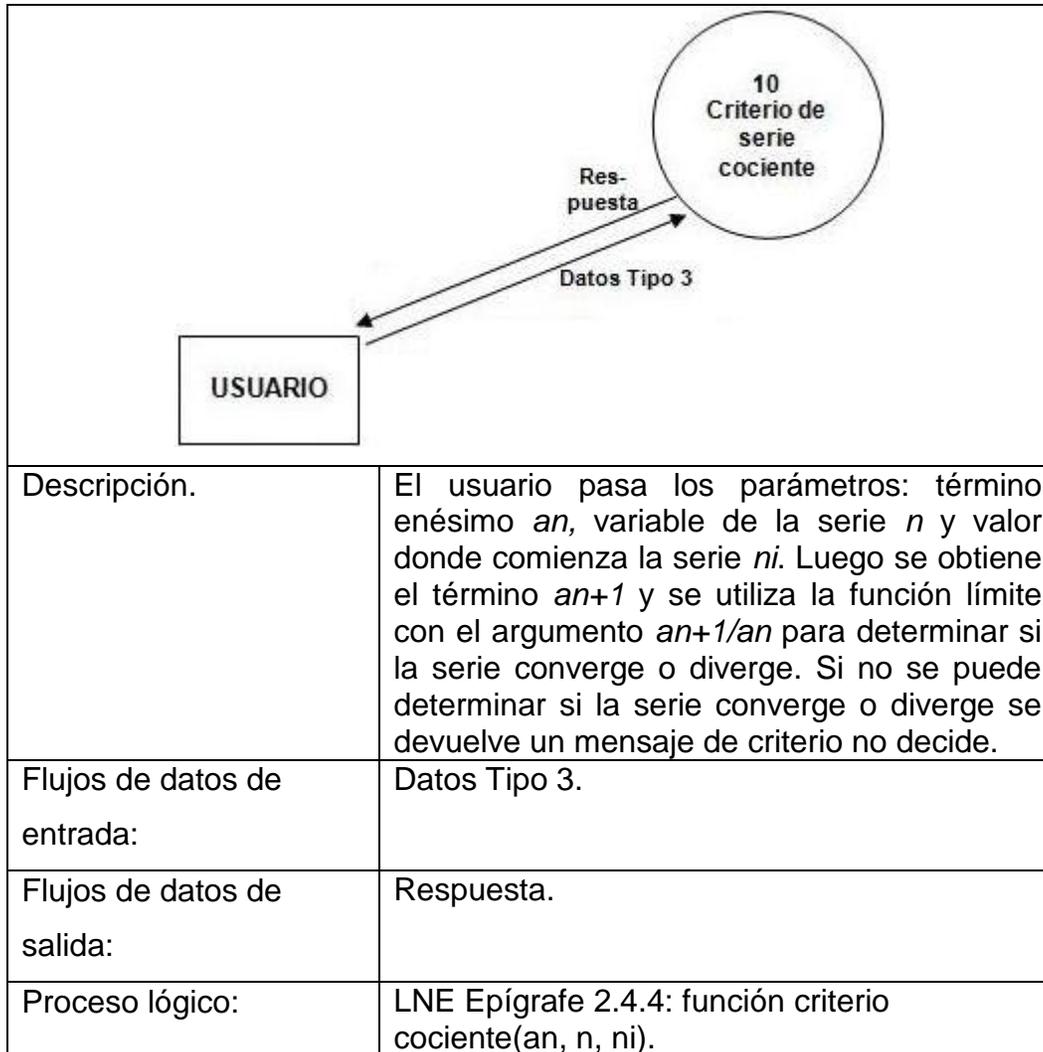


Tabla 21. Definición de Tarea Funcional. Criterio serie cociente.

Definición de Tarea Funcional	
Nombre de la tarea.	Criterio serie Raabe.
Número de identificación del DFD.	DFD 0.
Esquema.	

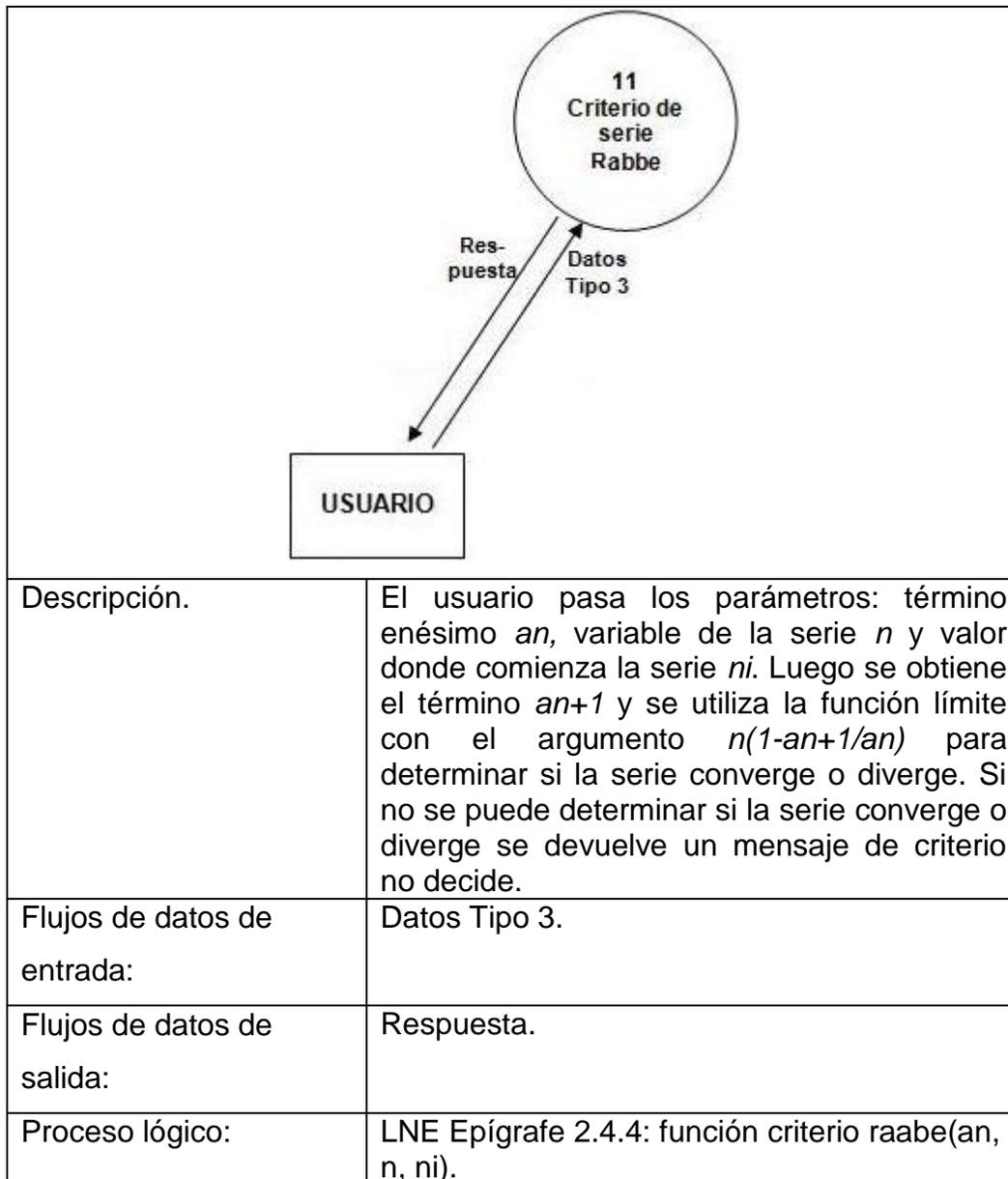


Tabla 22. Definición de Tarea Funcional. Criterio serie Raabe.

Definición de Tarea Funcional	
Nombre de la tarea.	Criterio serie Gauss.
Número de identificación del DFD.	DFD 0.

<p>Esquema.</p>	
<p>Descripción.</p>	<p>El usuario pasa los parámetros: término enésimo a_n, variable de la serie n y valor donde comienza la serie n_i. Luego se obtiene el término a_{n+1} y se expresa el cociente a_{n+1}/a_n como $1-Ln/n+Cn/n^2$ se utiliza la función límite y el valor Ln para determinar si la serie converge o diverge. Si no se puede determinar si la serie converge o diverge se devuelve un mensaje de criterio no decide.</p>
<p>Flujos de datos de entrada:</p>	<p>Datos Tipo 3.</p>
<p>Flujos de datos de salida:</p>	<p>Respuesta.</p>
<p>Proceso lógico:</p>	<p>LNE Epígrafe 2.4.4: función criterio gauss(a_n, n, n_i).</p>

Tabla 23. Definición de Tarea Funcional. Criterio serie Gauss.

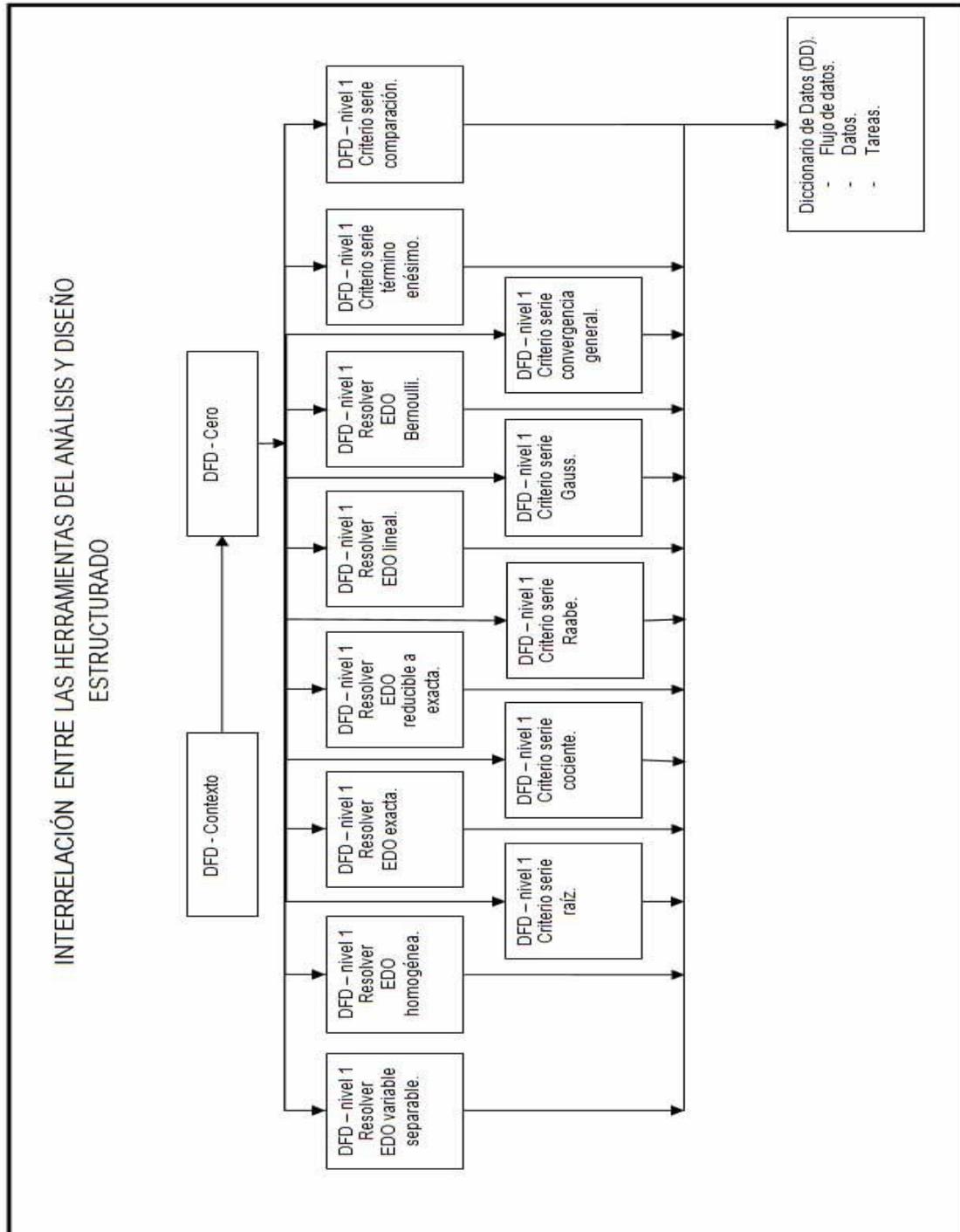
Definición de Tarea Funcional	
Nombre de la tarea.	Criterio serie convergencia.
Número de identificación del DFD.	DFD 0.

<p>Esquema.</p> <pre> graph TD Usuario[USUARIO] -- "Datos Tipo 3" --> Criterio((13 Criterio de serie convergencia)) Criterio -- "Respuesta" --> Usuario </pre>	
<p>Descripción.</p>	<p>El usuario pasa los parámetros: término enésimo a_n, variable de la serie n y valor donde comienza la serie n_i. Luego se evalúa el criterio del término enésimo. Si el criterio decide se devuelve su resultado, si no decide se evalúa el criterio de comparación. Si el criterio decide se devuelve su resultado, si no decide se evalúa el criterio de la raíz. Si el criterio decide se devuelve su resultado, si no decide se evalúa el criterio del cociente. Si el criterio decide se devuelve su resultado, si no decide se evalúa el criterio de Raabe. Si el criterio decide se devuelve su resultado, si no decide se evalúa el criterio de Gauss. Si ninguno de los criterios evaluados deciden se devuelve un mensaje de criterio no decide.</p>
<p>Flujos de datos de entrada:</p>	<p>Datos Tipo 3.</p>
<p>Flujos de datos de salida:</p>	<p>Respuesta.</p>
<p>Proceso lógico:</p>	<p>LNE Epígrafe 2.4.4: función criterio <code>comp_serie(an, n, ni)</code>.</p>

Tabla 24. Definición de Tarea Funcional. Criterio serie convergencia.

2.4.5 Relaciones entre las herramientas del análisis y diseño estructurado.

Las herramientas que forman parte de la metodología de análisis y diseño estructurado de sistemas y que han sido explicadas en los epígrafes anteriores, forman un conjunto perfectamente integrado. En el siguiente diagrama se detalla cómo utilizar todas estas herramientas interrelacionadas.



2.5 Conclusiones.

En el capítulo quedó definida la concepción general del sistema propuesto y los requerimientos mínimos de este para la utilización del sistema. Además, utilizando los métodos de análisis y diseño de sistemas estructurados, se detallaron los elementos que conforman esta metodología para la presente investigación: diagramas de flujo de datos, representación lógica en los procesos computacionales, diccionario de datos y la relación existente entre los mismos.

Capítulo III. Consideraciones para la utilización de la herramienta propuesta en la asignatura Matemática III

3.1 Introducción.

En este capítulo se realiza un análisis de una parte del Plan de Estudios D que sirve de base para comparar la herramienta Derive, utilizada actualmente, y Maxima, que se propone a utilizar. A través de esta comparación se obtienen ventajas del uso de la herramienta propuesta, basada en Maxima, así como criterios para considerar la utilización de la misma como instrumento que favorece el PDE de la asignatura Matemática III. Por último se presenta una propuesta de cómo utilizar la herramienta en la asignatura Matemática III.

3.2 Plan de Estudios D de la asignatura Matemática III en la carrera Ingeniería Informática.

A continuación, y para una mejor comprensión de este capítulo, se expone del Plan de Estudios D de la asignatura Matemática III en la carrera Ingeniería Informática (MES, 2007 pág. 125), los objetivos educativos, instructivos y el sistema de habilidades básicas a dominar.

Objetivos Educativos.

Contribuir a la formación de la concepción científica del mundo y del pensamiento científico mediante la comprensión de cómo se realiza un modelo matemático y como esto es un reflejo de la realidad.

Contribuir a que se desarrollen las capacidades cognoscitivas de los estudiantes, los hábitos de utilizar la literatura científica, la capacidad de razonamiento y del pensar lógicamente mediante el estudio de los temas de la asignatura.

Contribuir a la formación computacional de los estudiantes mediante la utilización de asistentes matemáticos, la creación de algunos programas y el desarrollo de la capacidad de algoritmizar.

Contribuir a la formación de la personalidad del alumno consolidando los hábitos de proceder reflexivamente y evaluar críticamente los resultados de su trabajo y la adecuación del método utilizado.

Objetivos Instructivos.

Desarrollar capacidades para caracterizar e interpretar los conceptos y teoremas más importantes relativos a las ecuaciones diferenciales y las Series.

Resolver modelos que se basen en ecuaciones diferenciales ordinarias y sistemas de ecuaciones diferenciales.

Modelar fenómenos de la realidad utilizando ecuaciones diferenciales ordinarias y sistemas de ecuaciones diferenciales.

Aproximar funciones por series, analizar las propiedades de la serie aproximadamente y utilizar la aproximación como instrumento para resolver problemas.

Habilidades básicas a dominar.

Identificar las ecuaciones diferenciales ordinarias (EDO) y parciales (EDP) y clasificarlas según orden y grado.

Interpretar los conceptos de solución de una ecuación diferencial.

Interpretar el teorema de existencia y unicidad para la solución de una EDO de primer orden.

Interpretar geoméricamente los diferentes conceptos de solución de EDO.

Identificar y clasificar por su tipo una EDO de primer orden y primer grado de Variables separables, Exactas, Lineales y de Bernoulli.

Identificar y clasificar Ecuaciones Diferenciales Lineales de Orden Superior.

Resolver EDO de primer orden y primer grado de Variables separables, Exactas, Lineales y de Bernoulli, y Ecuaciones Diferenciales Lineales de Orden Superior con Coeficientes Constantes y resolver problemas sencillos geométricos y vinculados con el perfil.

Utilizar asistentes matemáticos para resolver EDO.

Formular e interpretar el concepto de serie numérica, así como de serie convergente y serie divergente.

Determinar el carácter de una serie numérica.

Formular e interpretar el concepto de serie de funciones, así como de dominio de convergencia.

Obtener el dominio de convergencia de una serie de potencias.

Obtener el polinomio de Taylor de una función y el desarrollo en serie de Taylor de una función a partir de desarrollos conocidos y aplicarlo a problemas de aproximación de funciones.

Obtener el desarrollo en serie de Fourier de una función seccionalmente continua. Aplicar las series de Fourier a la solución de problemas sencillos de aproximación de funciones.

Utilizar asistentes matemáticos para resolver problemas utilizando series de potencias y series de Fourier.

3.3 Derive vs Maxima en la asignatura Matemática III.

A continuación se realiza una comparación entre el asistente Derive y Maxima, tomando como criterio, entre otros, las herramientas que presentan cada uno para apoyar el PDE de la asignatura Matemática III. Este análisis se realiza con el fin de fundamentar la utilización de la propuesta que se hace en esta investigación (Maxima, páginas webs e instrucciones programadas) como herramienta que favorece el cumplimiento de los objetivos del PDE de esta asignatura en el contexto del Plan de Estudios D analizado con anterioridad.

Derive vs. Maxima		
	Derive	Maxima
Descripción General	Herramienta que permite realizar operaciones matemáticas de forma simbólica. Presenta una interfaz convencional a modo de ventanas. Los objetos fundamentales que presenta son: variables, cadenas de caracteres, vectores, matrices, funciones, expresiones,	Herramienta que permite realizar operaciones matemáticas de forma simbólica. Presenta tres interfaces, una convencional a modo de ventanas, una interfaz gráfica a modo de comandos y una a consola a modo de comandos. Los objetos fundamentales que presenta son: variables, listas, cadenas de caracteres, vectores, matrices, funciones, expresiones,

	<p>ecuaciones y expresiones booleanas.</p> <p>Los módulos fundamentales que presenta son: aritmética, trigonometría, álgebra lineal, solución de ecuaciones, solución de sistemas de ecuaciones, cálculo diferencial e integral, lógica, programación lineal, gráficos en 2D y 3D.</p>	<p>ecuaciones y expresiones booleanas.</p> <p>Los módulos fundamentales que presenta son: aritmética, trigonometría, álgebra lineal, solución de ecuaciones, solución de sistemas de ecuaciones, cálculo diferencial e integral, lógica, programación lineal, gráficos en 2D y 3D, manejo de tensores, grafos, tratamiento de imágenes, exportación de resultados a Tex.</p>
Lenguaje de programación	Cuenta con un lenguaje de programación propio.	Cuenta con un lenguaje de programación propio. Además se puede programar en Lisp y Fortran.
Tipo de licencia que presenta	Copyright. Software privativo.	Copyleft. Software libre.
Sistemas operativos en los que se encuentra disponible	Windows.	Windows y GNU/Linux.
Soporte que lo respalda	<i>Texas Instrument</i> es la empresa que posee la licencia de Derive. Según la página oficial de esta empresa luego de la última distribución de Derive, realizada en el año 2003 (Derive 6), no desarrollará otra para este sistema.	Cuenta actualmente con el apoyo de una comunidad de desarrollo bajo el proyecto GNU máxima-source-forge.
	Tema I: EDO	
	<p>Instrucciones contenidas en los archivos <i>FirstOrderODEs.mth</i> y <i>SecondOrderODEs.mth</i>, las cuáles permiten resolver EDO de primer y segundo orden respectivamente.</p> <p>El paquete <i>FirstOrderODEs.mth</i> define instrucciones para resolver</p>	<p>Instrucciones para la solución de EDO, de sistema de EDO lineales, de problemas de valores iniciales y de contorno: <i>ode2</i>, <i>desolve</i>, <i>ic1</i>, <i>ic2</i>, <i>bc2</i> respectivamente.</p> <p>Además presenta funciones desarrolladas como parte de la herramienta propuesta tales como: <i>separable</i>, <i>homogénea</i>, <i>exacta</i>, <i>factor_integrante</i>,</p>

	<p>EDO de primer orden por los métodos: variables separables, homogéneas, exactas y lineales. El paquete <i>SecondOrderODEs.mth</i> define instrucciones para resolver EDO de segundo orden por los métodos: teorema fundamental en caso de EDO lineal y exactas</p>	<p>reducible_exacta, lineal_o1 y bernoulli.</p>
<p>Tema II: Series</p>		
<p>Herramientas, distribuidas por temas, que presenta para la asignatura Matemática III.</p>	<p>La instrucción o comando <i>sum</i> (\sum para el Derive) permite analizar la convergencia o determinar sumas de series numéricas y de funciones. La instrucción <i>Taylor</i>(exp, x, n) permite obtener el polinomio de <i>Taylor</i> de grado <i>n</i> de la expresión <i>exp</i>.</p>	<p>Las instrucciones o comandos <i>sum</i> y <i>simpsum</i> permiten obtener sumas de series numéricas y de funciones.</p> <p>La instrucción <i>taylor</i> obtiene un polinomio formado por los primeros términos de una serie de potencias La instrucción <i>powerseries</i> obtiene el desarrollo general o los primeros términos de una serie de potencias.</p> <p>Las instrucciones <i>fourier</i>, y <i>totalfourier</i> permiten obtener los coeficientes o el desarrollo general en una serie de Fourier.</p> <p>Además presenta funciones para evaluar la convergencia de series numéricas desarrolladas como parte de la herramienta propuesta tales como:</p> <p><i>termino_enesimo</i>, <i>comparacion</i>, <i>raiz</i>, <i>cociente</i>, <i>raabe</i>, <i>gauss</i>, <i>convergencia_general</i>.</p>
<p>Documentación específica para la asignatura Matemática III.</p>	<p>No presenta ningún tipo de documentación orientada a la asignatura. En la ayuda del sistema se puede encontrar información acerca de las instrucciones de los archivos <i>FirstOrderODEs.mth</i> y <i>SecondOrderODEs.mth</i> utilizadas en el Tema I</p>	<p>Presenta una página web realizada específicamente para la asignatura organizada por temas con instrucciones para cada uno de los contenidos de la misma.</p> <p>Además de esta página web titulada <i>matematicaIII.html</i> presenta dos páginas web: una de ejemplos</p>

	(EDO).	<i>matematicaIII_ejemplos.html</i> y otra con el código fuente de todas las instrucciones desarrolladas en la herramienta <i>matematicaIII_codigos.html</i> .
--	--------	--

Tabla 25. Derive vs. Maxima

3.4 Ventajas de la utilización de la herramienta propuesta en la asignatura Matemática III.

A continuación se exponen ventajas de utilizar la herramienta propuesta, basada en Maxima, en la asignatura Matemática III:

1. Maxima es un sistema con licencia libre.
2. Maxima cuenta en la actualidad con una comunidad de desarrollo bajo el proyecto maxima-source-forge que sirve de soporte para la implementación de las nuevas distribuciones.
3. Maxima cuenta con versiones para los sistemas operativos Windows y GNU/Linux.
4. Maxima cuenta con una extensa documentación en artículos, manuales y libros.
5. Maxima cuenta con un lenguaje propio de programación. Además soporta programación en Lisp y Fortran. Esto permite un mayor vínculo con la especialidad de la carrera Ingeniería Informática.
6. Maxima cuenta con instrucciones para resolver EDO y sistema de EDO por métodos analíticos e instrucciones para obtener sumas de series numéricas y de funciones y para representar funciones mediante series de potencias y trigonométricas (Fourier). Estas instrucciones guardan estrecha relación con los contenidos de la asignatura Matemática III abordados en el Plan de Estudios D.
7. Como parte de la herramienta desarrollada se implementaron instrucciones (en y para Maxima) para completar los contenidos de la asignatura Matemática III, potenciando los diferentes métodos de solución de EDO y la evaluación de la convergencia de series numéricas mediante los diferentes criterios existentes.

8. La herramienta desarrollada contempla una documentación en forma de guía para la explotación de Maxima con las instrucciones propias del sistema y las implementadas.
9. Parte de la documentación de la herramienta desarrollada consiste en la publicación del código fuente de todas las instrucciones implementadas, lo que permite un estudio por parte de los estudiantes de métodos teóricos vistos en el aula y ahora programados. Esto motiva a los estudiantes de esta especialidad a mejorar instrucciones ya existentes e implementar nuevas.

3.5 Herramienta propuesta como instrumento que favorece el PDE de la asignatura Matemática III.

A continuación se presenta la fundamentación de la herramienta propuesta, basada en Maxima, como instrumento que favorece el PDE de la asignatura Matemática III a través del cumplimiento de los objetivos y el desarrollo de habilidades del Plan de Estudios D.

De los objetivos educativos:

La utilización de una herramienta como la propuesta, basada en software libre, contribuye a la formación de la personalidad del alumno al fomentar valores éticos, de solidaridad y libertad. Además la herramienta propuesta, basada en Maxima, permite el procesamiento de modelos matemáticos complejos (de EDO, sistema de EDO y series) que reflejen situaciones reales. En las actividades docentes, en aulas, los modelos utilizados son generalmente simplificados y llevados a un ámbito educativo, es por ello que la utilización de la herramienta propuesta en el procesamiento de modelos matemáticos que reflejen situaciones reales contribuye a la formación del pensamiento científico.

Contar con una herramienta que puede ser estudiada para conocer su funcionamiento contribuye a desarrollar las capacidades del conocimiento. El estudio de la amplia bibliografía de Maxima permite fomentar hábitos de utilizar la literatura científica en los estudiantes, mientras que el estudio de la guía desarrollada en páginas web para los diferentes temas de la asignatura como

parte de esta investigación favorece el pensamiento lógico. La publicación del código fuente de instrucciones programadas estimula a los estudiantes al estudio de los algoritmos de las mismas. En las actividades de laboratorio al dejar propuesta la implementación de instrucciones no programadas en la herramienta (Ejemplo: implementación de instrucciones para la obtención de la solución de EDO de Euler-Cauchy y/o para la obtención del intervalo de convergencia de series de potencias) se contribuye a la formación computacional de los estudiantes mediante la creación de programas y el desarrollo de la capacidad de algoritmizar.

De los objetivos instructivos:

La herramienta propuesta presenta instrucciones para resolver modelos que se basen en EDO, sistema de EDO y para aproximar funciones mediante series, esto favorece de manera directa los objetivos instructivos declarados en el plan de estudios.

Del sistema de habilidades básicas a dominar:

Al utilizar las instrucciones `ode2` (propia de Maxima), `separable_gen`, `homogenea_gen`, `exacta_gen`, `reducible_exacta_gen`, `lineal_o1_gen` y `bernoulli_gen` (desarrolladas) para resolver modelos de EDO se logra una mejor interpretación de los conceptos de solución de una ecuación diferencial, además que favorece la identificación y clasificación de EDO de primer orden en variables separables, exactas, lineales y de Bernoulli. De igual forma, el utilizar la instrucción `desolve` (propia de Maxima) permite resolver EDO lineales de orden superior con coeficientes constantes. El desarrollo de la habilidad “utilizar asistentes matemáticos para resolver EDO” está incluido en la explicación anterior.

Al utilizar las instrucciones `termino_enesimo`, `comparacion`, `raiz`, `cociente`, `raabe`, `gauss` y `comp_serie` (desarrolladas) para determinar, si es posible, el carácter de una serie numérica (convergente o divergente) se logra una mejor interpretación del concepto de serie numérica, así como de serie convergente y serie divergente.

La instrucción `powerseries` (propia de Maxima) fomenta el desarrollo de habilidades como: formular e interpretar series de funciones y formular e interpretar series de potencias. La instrucción `taylor` (propia de Maxima) permite el

desarrollo de la habilidad: obtener el polinomio de Taylor a partir de una función así como el desarrollo en serie de Taylor de una función a partir de desarrollos conocidos y aplicarlo a problemas de aproximación de funciones. De igual forma al utilizar las instrucciones `fourier` y `totalfourier` (propias de Maxima) se favorece el desarrollo de la habilidad: obtener el desarrollo en serie de Fourier de una función seccionalmente continua y aplicar las series de Fourier a la solución de problemas sencillos de aproximación de funciones. De esta manera también se desarrolla la habilidad “utilizar asistentes matemáticos para resolver problemas utilizando series de potencias y series de Fourier”.

Luego de haber fundamentado la herramienta propuesta como instrumento que favorece el PDE de la asignatura Matemática III mediante el cumplimiento de los objetivos educativos, instructivos y del sistema de habilidades del Plan de Estudios D, se presenta una propuesta de cómo utilizar la herramienta en la asignatura Matemática III:

En actividades docentes en el aula:

En el desarrollo del tema #1 (EDO), dejar propuesto la utilización de la herramienta (que comprende el estudio de la guía y la utilización de las instrucciones) para obtener la solución de modelos de EDO. A medida que se avanza en el estudio de los diferentes métodos para la solución de EDO de primer orden ir vinculando cada método con la respectiva instrucción que se encuentra en la herramienta. Ejemplo: cuando se estudie el método de variable separable dejar propuesto la utilización de la instrucción `separable_gen`, que implementa este método, prestando especial atención a la explicación de los parámetros de la instrucción. En este caso una función $p(x)$ y otra función $q(y)$ que conforman la EDO $\frac{dy}{dx} = p(x) \cdot q(y)$. Realizar de esta forma la vinculación entre el contenido y la herramienta desarrollada motiva al estudiante a utilizarla y a entender mejor cada método.

En el desarrollo del tema #2 (Series) proceder de manera similar a como se explicó en el tema #1.

En actividades de laboratorio:

Las actividades de laboratorio se clasifican en laboratorios de familiarización y laboratorios evaluativos. En la asignatura Matemática III todas las actividades de laboratorio son de familiarización. El laboratorio de familiarización consiste en una actividad donde el estudiante a partir de los métodos o conocimientos adquiridos en las actividades docentes en el aula y de una explicación inicial por parte del profesor, utiliza el asistente matemático (hasta ahora siempre se ha utilizado Derive) para resolver ejercicios vistos en clases y comprobar el resultado así como resolver ejercicios propuestos o ejercicios de gran magnitud no abordados en clases de pizarra y tiza. El profesor en este tipo de actividad es el encargado de evacuar cualquier duda que tenga el estudiante, ya sea del ejercicio que está resolviendo o de cómo utilizar el asistente matemático en ese ejercicio.

La herramienta propuesta, cuenta con una guía (página web) orientada específicamente a los contenidos de los temas de la asignatura. El profesor debe orientar los laboratorios de familiarización por esta guía.

En las actividades de laboratorio el profesor, puesto que parte de la herramienta consiste en la publicación del código fuente de las instrucciones desarrolladas y se puede acceder de igual manera el código fuente de las instrucciones propias de Maxima, debe fomentar en los estudiantes el estudio del código fuente para que conozcan la programación de los métodos teóricos estudiados y la mejoren o modifiquen. En los laboratorios del tema #1 se deben dejar propuesto la programación de nuevas instrucciones como: instrucción para obtener la solución de EDO de Euler-Cauchy o instrucción para resolver EDO lineales con coeficientes constantes por el teorema fundamental. En los laboratorios del tema #2 se deben dejar propuesto la programación de instrucciones como: instrucción para evaluar la convergencia de una serie numérica mediante el criterio de la integral o instrucción para obtener el intervalo de convergencia de series de potencias.

3.6 Conclusiones.

En este capítulo se analiza la influencia de los programas Derive y Maxima como herramientas que permiten el cumplimiento de los objetivos y sistemas de habilidades en la asignatura Matemática III en la carrera Ingeniería Informática.

Como aspecto importante se señala que Maxima es un software libre mientras que Derive (programa utilizado actualmente en la asignatura) no lo es. Derive carece de instrucciones para una mejor explotación del Tema II: Series, de la asignatura Matemática III mientras que Maxima cuenta con instrucciones (propias del sistema y desarrolladas) que permiten una mejor explotación. La herramienta informática `matematicaIII.html` desarrollada para ser utilizada en Maxima, le facilita al estudiante una mayor comprensión de los temas abordados en la asignatura y al mismo tiempo le sirve como material auxiliar en el aprendizaje, validación e interpretación de los mismos.

Conclusiones.

1. La utilización de asistentes matemáticos basados en la filosofía de software libre enriquece el PDE de la asignatura Matemática III de la disciplina Matemática General de la carrera Ingeniería Informática.
2. El asistente matemático Maxima permite el desarrollo de una herramienta que se adapte a los objetivos del PDE de la asignatura Matemática III, planteados en el Plan de Estudios D de la carrera Ingeniería Informática.
3. La herramienta desarrollada favorece el cumplimiento de los objetivos del PDE de esta asignatura.

Recomendaciones.

1. Utilizar la herramienta desarrollada en este trabajo para el desarrollo del PDE de la asignatura Matemática III en la carrera Ingeniería Informática.
2. Utilizar, por parte de los estudiantes de la carrera Ingeniería Informática, el asistente matemático Maxima para desarrollar algoritmos matemáticos que no estén implementados.
3. Utilizar asistentes matemáticos basados en la filosofía de software libre para enriquecer el PDE de la asignatura Matemática III de la disciplina Matemática General de la carrera Ingeniería Informática.
4. Analizar la posibilidad de extender la utilización de esta herramienta en otras universidades.

Bibliografía.

Acuña, Karenny Brito. 2009. *Selección de Metodologías de Desarrollo para Aplicaciones Web en la Facultad de Informática de la Universidad de Cienfuegos.* Informática. Facultad de Informatica, Universidad de Cienfuegos. Cienfuegos : s.n., 2009.

Asistentes matemáticos y Software libre. **Medina, Juan Felipe. 2008.** Cienfuegos : s.n., Diciembre de 2008, Anuario de la Universidad "Carlos Rafael Rodríguez", pág. 10. 978-959-257-195-0.

autores, varios. 2003. *Axiom.* 2003. Vol. 0.

Axiom. Axiom book. [En línea] [Citado el: 10 de diciembre de 2008.]

<http://page.axiom-developer.org/zope/Plone/refs/books/axiom-book2.pdf>.

Blanco Encinosa, Lázaro J. y Gutztat, Ida R. 1991. *Sistemas Informáticos. Teoría, métodos de elaboración, técnicas y herramientas.* Ciudad Habana : Universidad de La Habana, 1991. Vol. 1.

Cañedo, Carlos. 2008. *El diseño curricular en la educación superior.* Cienfuegos : s.n., 2008.

Coca, Yandira Mouriz. 2005. *Registro de Enfermedades de Declaración Obligatoria para el Sistema Integral de Salud.* 2005.

Daniele, Marcela. 2007. Análisis y Diseño de Sistemas (3303). [En línea] 2007. http://dc.exa.unrc.edu.ar/nuevodc/materias/sistemas/2007/TEORICOS/TEORIA_1.

Eaton, John W. 2008. *GNU Octave. A high-level interactive language for numerical computations.* 2008.

Euler sourceforge. [En línea] [Citado el: 12 de diciembre de 2008.]

<http://euler.sourceforge.net/index.html>.

Galván, J. Rafael Rodríguez. 2005. *Matemáticas y Software libre para la docencia en la Universidad de Cádiz.* Cádiz : s.n., 2005. pág. 20.

—. **2007.** *Maxima con wxMaxima: software libre en el aula de matemáticas.* Cádiz : s.n., 2007.

Guzmán, Christian Peralta. 2009. *Entra Cuba al mundo del software libre.* [En línea] 2009. [Citado el: 8 de Mayo de 2009.]

<http://colectivoquimica.dnsalias.org/Members/admin/entra-cuba-al-mundo-del-software-libre>.

Letelier, Patricio. "Proceso de desarrollo de software". [En línea]

<http://pid.dsic.upv.es/C1/Material/Documentos%2520Disponibles/Introducci%C3%B3n%2520Proceso%2520de%2520Desarrollo%2520de%2520SW.doc>.

Lossen, H. Schönemann. 2006. *21 Years of Singular Experiments in Mathematics*. 2006.

Marco, T. De. 1979. *Structured Analysis and Systems Specification*. New Jersey : s.n., 1979.

Medina, Juan Felipe. 2008. "OPTIVACORTUNI. Sistema informático para resolver problemas de optimización en procesos de corte de materiales". Informática, Universidad de Cienfuegos. Cienfuegos : s.n., 2008. pág. 100.

MES. 2007. *Plan de estudio D. Ingeniería Informática Presencial*. 2007.

Paradis, Emmanuel. 2002. *R para principiantes*. Montpellier : Universit Montpellier, 2002.

Pardini, A. 2007. *Fundamentación del uso de software libre en la Universidad Pública. Enseñando matemática con herramientas alternativas*. 2007.

s.n. Las Personas en las Metodologías de Ingeniería del Software. [En línea] <http://www26.brinkster.com/lwelicki/articles/PersonasMetodologiaJIS04.pdf>.

Schenck, Hal. 2003. *Computational Algebraic Geometry*. 2003.

Stallman, Richard M. 2004. *Software libre para una sociedad libre*. Madrid : Traficantes de Sueños, 2004. pág. 232. 84-933555-1-8.

Weinberg, V. 1978. *Structured Analysis*. New Jersey : s.n., 1978.

Wikipedia Macaulay. Macaulay - Wikipedia, la enciclopedia libre. [En línea] [Citado el: 15 de diciembre de 2008.] http://en.wikipedia.org/wiki/Macaulay_2.

Wikipedia Maxima. Maxima-Wikipedia, la enciclopedia libre. [En línea] [Citado el: 12 de diciembre de 2008.] <http://es.wikipedia.org/wiki/Maxima>.

Wikipedia Singular. Singular-Wikipedia, la enciclopedia libre. [En línea] [Citado el: 15 de diciembre de 2008.] <http://es.wikipedia.org/wiki/SINGULAR>.

Wikipedia, Enciclopedia. Derive - Wikipedia, la enciclopedia libre. [En línea] [Citado el: 2008 de noviembre de 2008.] <http://es.wikipedia.org/wiki/Derive>.

Yacas. Yacas. [En línea] [Citado el: 10 de diciembre de 2008.]

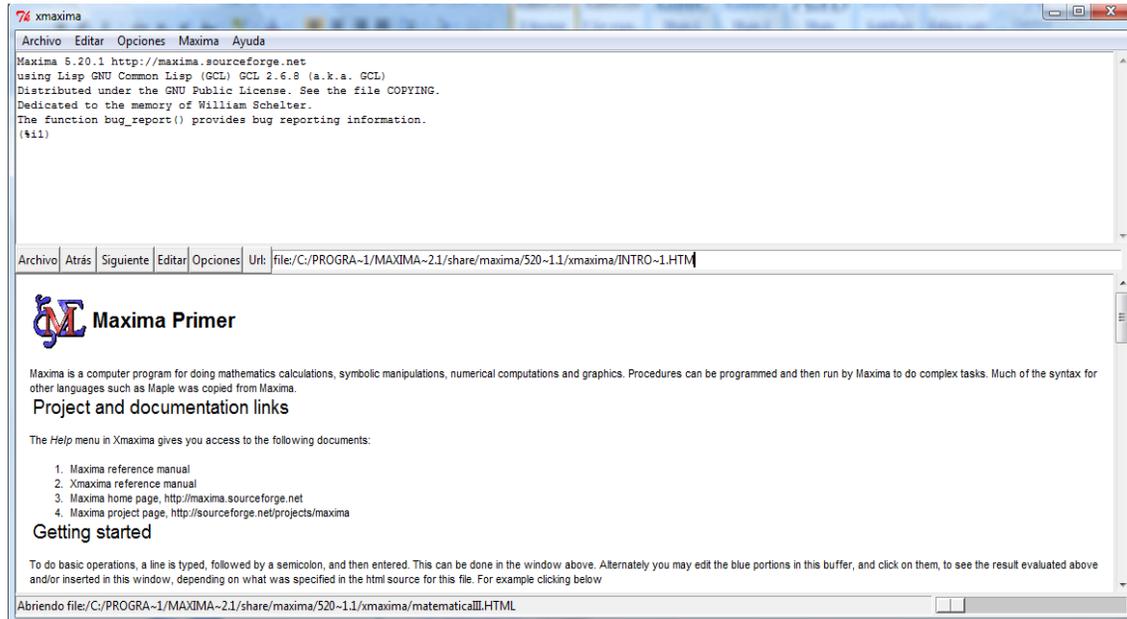
<http://yacas.sourceforge.net>.

Yourdon. 1982. *Controlling Software Projects*. Press : s.n., 1982.

Yurramendi, Yosu. 2007. *Curso básico de R*. Cienfuegos : s.n., 15 de febrero de 2007.

Anexos.

Anexo 1. Ventana principal de xMaxima.



Anexo 2. Ventana de xMaxima con la página web matematicaIII.html cargada.

